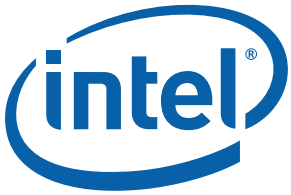


Intel® Communications Chipset 89xx Series

Datasheet

April 2014

Order Number: 327879-005US

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Intel Corporation may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights that relate to the presented subject matter. The furnishing of documents and other materials and information does not provide any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

Designers must not rely on the absence or characteristics of any features or instructions marked “reserved” or “undefined.” Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at http://www.intel.com.

Intel, the Intel logo and Pentium are registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2014, Intel Corporation. All Rights Reserved.

Revision History

Date	Revision	Description
March 2014	005	• Updated Section 1.1 Introduction • Updated Table 1-1 added DH8926CL. • Updated Section 1.2 added bullets. • Added Section 1.2. Differences Between DH89xxCC and DH89xxCL SKUs. • Added Table 1-2. Differences Between DH89xxCC and DH89xxCL SKUs. • Correction to Table 3-3. • Clarification to Table 5-4. • Clarified note, Section 6.9.1.4. • Correction to Table 6-137 through Table 6-139 and Table 6-142. • Correction to Table 8-1 and Table 8-3. • Added table footnote to Table 9-3. • Added table footnote to Table 10-1 • Corrected to Table 10-3. • Added table footnote to Table 11-1. • Corrected table Table 11-3. • Added table footnote to Table 17-1. • Corrected table Table 17-3. • Corrected note, Table 19-3. • Correction to Table 20-23. Corrected SKU values & Added DH8926CL. • Added clarifications to Table 32-1. • Correction to Table 32-2. • Added Table 32-22 Footnote 5 and Table Note. • Added Table 33-2 Footnote for DH89xxCC maximum voltage. • Corrected Figure 33-19. GBE_EE_DO & GBE_EE_DI were reversed. • Added Chipset references 8903, 8910, 8920, 8925, 8950, and 8955 to Section 33.0.
December 2013	004	• Updated Table 3-3. Added DH89xxCL Device IDs. • Updated Table 3-4. Added DH89xxCL Device IDs. • Updated Section 19.2.3.1.
December 2013	003	• Updated Table 1-1. Added DH8925CL, DH8950CL & DH8955CL SKUs. • Updated Figure 2-2 to include the Added DH8925CL, DH8950CL & DH8955CL features. • Updated Table 2-1 to include the Added DH8925CL, DH8950CL & DH8955CL features. • Correction to Table 4-4. • Updated Table 4-22. Changed “CPUSCI_STS” to DMISCI_STS”. • Updated Table 4-22. Changed “OS_TCO_SMI” to “SW_TCO_SMI”. • Updated Section 4.12.3.2. • Updated Section 6.6.1 Changed “he” to “the”. • Updated Table 7-65. Changed “0062003h” to “06200003h”. • Updated Table 7-66. Changed bit field description. • Updated Table 12-49. Changed “DMI” to “Root Port”. • Updated Section 20.2.2.18. • Updated Chapter 21.0. Added note in the introduction. • Corrected typo in Section 22.5.2.2. • Updated Section 24.11.6 • Updated Section 28.7.1.23. Changed “Section 28-62” to “Table 28-62”. • Updated Chapter 32.0 to include the Added DH8925CL, DH8950CL & DH8955CL.

Date	Revision	Description
		Continued
July 2013	002	• Updated Table 3-3. Deleted "SATA* Controller 1 Desktop: in IDE, supporting 2 ports; DID = 0x2321". Added operational support details for SATA* Controller #1 and SATA* Controller #2. • Updated Section 4.10.1.5 and Table 4-32. Changed "PROCPWRGD" to "CPUPWRGD" • Updated Section 4.15. Changed "ACHI" to "AHCI" • Updated Section 4.5.1. Changed "ACHI" to "AHCI"; and added PCIe* B:D:F mapping details for SATA* Controllers #1 and #2. • Updated Section 4.15.7. Changed Section Title from "ATA LED" to "SATA* LED" • Updated Figure 4-22. Changed "ECHI" to "EHCI" • Updated Section 10.2.2.9. Removed Port(x) Offset Address Mapping for when RMH is disabled since RMH is always enabled. • Updated Table 12-4; Offset 04h: PCI COMMAND Register. Re-assigned Register Bits to sync with PCIe* Specification Register Bit definitions. • Updated Table 20-10 Assigned the correct Major and Minor Revision default values for the chipset. • Updated Table 21-1. Refined the definition of EEPROM. • Updated Section 21.5. Added a Note to define "FLEEP" • Updated Section 22.5.2.2.2 Added wording to indicate that Ownership Acquisition of Shared Resources is required before MDIC Read/Write cycles can be executed. • Updated Section 22.5.6.2.1 Added correct setting of CTRL_EXT.LINK_MODE for MAC Loopback. • Updated Section 24.3.12 "LED 0 Configuration Defaults (LAN Base Address + Offset 0x1F)". Updated cross-references for the EEPROM Word • Updated Section 24.3.13"Software Defined Pins Control (LAN Base Address + Offset 0x20)". Added Description Clarification Notes to Bit[11] and Bit[10] • Updated Table 28-9 In the Note for For Bit[7], removed reference to LAN_PWR_GOOD. • Updated Table 28-14. New definition for Bit[31] = '0b'. • Updated Table 28-15. All bits are "Reserved", except Bit[2]. • Updated Section 28.17.1.8 Added Table 28-184, "SFP I2C Command - I2CCMD [0:3] (0x1028; R/W)" • Updated Table 32-8. Specification Update for GBE[0:3]_LED configuration straps. External strap requirement. • Updated Table 32-22. Changed "Native" Default Mode to "GPIO" • Updated Table 32-27. Updated strap requirements for GBE[0:3]_LED Signals • Updated Table 32-29 Updated Description for VCC3P3_RTC • Updated Table 33-2. Updated VCC3P3_RTC and VCC3P3_RTC (Battery) min/max Power Rail values. Added Note for VCC3P3_RTC voltage requirement specification. • Updated Table 33-3. Removed Table Note 1- "Pre-silicon estimates and subject to change" • Updated Table 33-9. Updated IOL(max) for VOL7 and VOL8 (JTDO / EP_JTDO Output Low) • Updated Table 33-25. Removed AC coupling capacitor for GbE Clock (GBE_CLK100[P,N]) • Updated Table 33-29. Updated the minimum (min) values for t188a (SPI_CLK high time) and t189a (SPI_CLK low time) for SPI Timings (20MHz).
August 2012	001	• Updated Table 33-1 through Table 33-5.

Product Features

- Platform Controller Hub (PCH)
  - — Integrated Intel® Platform Controller Hub Complex technology
  - — Intel Architecture Processor Companion
  - — Asynchronous DIMM Refresh (ADR)
  - — Intel® QuickAssist Technology
  - — Extensive integration of standard Intel architecture communications interfaces provide cost, power and board area savings
- Intel® QuickAssist Integrated Accelerator
  - — Symmetric Cryptographic Functions
  - — Public Key Functions
  - — Compression/Decompression
- Direct Media Interface (DMI) Gen1
  - — 10 Gb/s each direction, full duplex
  - — Transparent to software
- PECI Interface
- PCI Express* Gen1
  - — 4 PCI Express* Root Complex Ports
  - — PCIe* Gen1 speed (2.5GT/s)
  - — Compliant to Gen2 messaging protocol
  - — Ports can be independently configured to support 4x1, 2x2, 1x2 + 2x1, or 1x4
  - — Supports lane reversal with x4 configuration
  - — Module based Hot-Plug supported (for example, ExpressCard*)
- Integrated Serial ATA Host Controller
  - — Two SATA* ports
  - — SATA* Gen2 Data transfer rates up to 3.0 Gb/s (300 MB/s).
  - — One activity LED
  - — Multiple MSI Message vectors.
  - — Integrated AHCI controller
- USB* 2.0/1.1
  - — Six USB* 1.1 or USB* 2.0
  - — One EHCI Host Controller, supporting up to six external ports.
  - — Per-Port-Disable Capability
  - — Includes two USB* 2.0 High-speed Debug Ports
  - — Supports wake-up from sleeping states S1-S4
  - — Supports legacy Keyboard/Mouse software
- UART
  - — Two integrated UARTs 16550 compatible
- SMBus
  - — SMBus Interfaces
  - — One Host SMBus (Master)
  - — One SMLINK (Slave)
  - — One EndPoint SMBus (Slave)
  - — One GbE SMBus (Master/Slave)
  - — SMBus Max speed, up to 100 KHz
  - — Supports SMBus 2.0 Specification
  - — Host interface allows processor to communicate via SMBus
  - — Slave interface allows an internal or external microcontroller to access system resources
  - — Compatible with most two-wire components that are also I2C compatible
- High Precision Event Timers
  - — Advanced operating system interrupt scheduling
- Timers Based on 82C54
  - — System timer, Refresh request, Speaker tone output
- Real-Time Clock
  - — 256-byte battery-backed CMOS RAM
  - — Integrated oscillator components
  - — Lower Power DC/DC Converter implementation
- System TCO Reduction Circuits
  - — Timers to generate SMI# and Reset upon detection of system hang
  - — Timers to detect improper processor reset
  - — Integrated processor frequency strap logic
  - — Supports ability to disable external devices
- Serial Peripheral Interface (SPI)
  - — Supports up to two SPI devices
  - — Two Chip Select pins, up to 16MB per memory device.
  - — Supports 20/33/50 MHz SPI devices.
  - — Support up to two different erase granularities.
- Interrupt Controller
  - — Supports SERIRQ interrupt pin
  - — Supports PCI 2.3 Message Signaled Interrupts
  - — Two cascaded 82C59 with 15 interrupts
  - — Integrated I/O APIC capability with 24 interrupts
  - — Supports Processor System Bus interrupt delivery
- DMA Controller
  - — Two cascaded 8237 DMA controllers
  - — Supports LPC DMA

- Power Management Logic
  - — Supports ACPI 3.0b
  - — ACPI-defined power states (processor driven C states)
  - — ACPI Power Management Timer
  - — SMI# generation
  - — All registers readable/restorable for proper resume from 0 V suspend states
- Support for APM-based legacy power management for non-ACPI implementations
- Integrated Gigabit Ethernet Controllers
  - — Four Integrated IEEE 802.3 MACs
  - — Four SGMII/SerDes interface Ports
  - — Four I2C/MDIO Ports for External PHY configuration[1]
  - — Serial EEPROM interface
  - — 10/100/1000 Mbps Ethernet Support
  - — Jumbo Frame Support
- PCI Express* Gen2 End Point
  - — SR-IOV support for Intel® QuickAssist Technology
  - — PCI Express* 2.0 specification running at 5GT/s.
  - — Supports lane reversal
- Low Pin Count (LPC) I/F
  - — Supports two Master/DMA devices
  - — Support for Security Device (Trusted Platform Module) connected to LPC
- 68 GPIO pins (multiplexed or dedicated)
  - — TTL, Open-Drain, Inversion
  - — GPIO lock down
- Package
  - — 27 mm x 27 mm FCBGA
  - — 942 pin
- JTAG 1149.1
  - — Boundary Scan for testing during board manufacturing

1. *See the Supported Ethernet PHY Devices for the Intel® Communications Chipset 89xx Series Application Note* for more information.

Contents

Intel® Communications Chipset 89xx Series 1 Overview and PCH Interfaces - Volume 1 of 4 82

1.0 Introduction 84
1.1 Introduction 84
1.2 Intel® Communications Chipset 89xx Series SKU Definition 85
1.3 Differences Between DH89xxCC and DH89xxCL SKUs 86
1.4 Document Organization 87
1.5 Referenced Documents and Related Websites 88
1.6 Acronyms 89
1.7 Glossary 92

2.0 Architecture Overview 97
2.1 Introduction 97
2.2 PCH Architecture Overview 99
2.2.1 PCH Block Summary 100
2.2.2 PCH External Interfaces 100
2.2.3 IA Compatibility 101

3.0 PCH Platform Memory and Device Configuration 102
3.1 Overview 102
3.1.1 Configuration Objectives 102
3.1.2 Terminology and Conventions 103
3.2 IA Platform Infrastructure 103
3.2.1 General IA Platform View of the Physical Address Space 103
3.2.2 IA Platform View of Configuration 103
3.3 High-Level Views 104
3.3.1 Characteristics of External System Memory (DRAM) 104
3.3.2 Characteristics of Internal and External Memories 104
3.3.3 Characteristics of Device Configuration 105
3.4 Memory Map for PCIe* Endpoint-Attached Devices 105
3.5 PCH Endianness 106
3.6 PCI Configuration 106
3.6.1 Overview 106
3.6.2 Device Tree 107
3.6.3 Materializing Device Structures 109
3.6.4 PCI Configuration Headers 109

4.0 Functional Description 112
4.1 PCI Express* Root Ports 112
4.1.1 Interrupt Generation 112
4.1.2 Power Management 113
4.1.2.1 S3/S4/S5 Support 113
4.1.2.2 Resuming from Suspended State 113
4.1.2.3 Device Initiated PM_PME Message 113
4.1.2.4 SMI/SCI Generation 114
4.1.3 SERR# Generation 114
4.1.4 Hot-Plug 114
4.1.4.1 Presence Detection 114
4.1.4.2 Message Generation 114
4.1.4.3 Attention Button Detection 115
4.1.4.4 SMI/SCI Generation 115
4.2 LPC Bridge (with System and Management Functions) (B0:D31:F0) 116

4.2.1 LPC Interface 116
4.2.1.1 LPC Cycle Types 117
4.2.1.2 Start Field Definition 117
4.2.1.3 Cycle Type / Direction (CYCTYPE + DIR) 118
4.2.1.4 Size 118
4.2.1.5 SYNC 118
4.2.1.6 SYNC Time-Out 119
4.2.1.7 SYNC Error Indication 119
4.2.1.8 LFRAME# Usage 119
4.2.1.9 I/O Cycles 119
4.2.1.10 Bus Master Cycles 119
4.2.1.11 LPC Power Management 120
4.2.1.12 Configuration and Implications 120
4.3 DMA Operation (B0:D31:F0) 120
4.3.1 Channel Priority 121
4.3.1.1 Fixed Priority 121
4.3.1.2 Rotating Priority 121
4.3.2 Address Compatibility Mode 121
4.3.3 Summary of DMA Transfer Sizes 122
4.3.3.1 Address Shifting When Programmed for 16-Bit I/O Count by Words 122
4.3.4 Autoinitialize 122
4.3.5 Software Commands 123
4.4 LPC DMA 123
4.4.1 Asserting DMA Requests 123
4.4.2 Abandoning DMA Requests 124
4.4.3 General Flow of DMA Transfers 124
4.4.4 Terminal Count 125
4.4.5 Verify Mode 125
4.4.6 DMA Request Deassertion 125
4.4.7 SYNC Field / LDRQ# Rules 126
4.5 8254 Timers (B0:D31:F0) 126
4.5.1 Timer Programming 127
4.5.2 Reading from the Interval Timer 128
4.5.2.1 Simple Read 128
4.5.2.2 Counter Latch Command 128
4.5.2.3 Read Back Command 128
4.6 8259 Interrupt Controllers (PIC) (B0:D31:F0) 129
4.6.1 Interrupt Handling 130
4.6.1.1 Generating Interrupts 130
4.6.1.2 Acknowledging Interrupts 130
4.6.1.3 Hardware/Software Interrupt Sequence 130
4.6.2 Initialization Command Words (ICWx) 131
4.6.2.1 ICW1 131
4.6.2.2 ICW2 131
4.6.2.3 ICW3 132
4.6.2.4 ICW4 132
4.6.3 Operation Command Words (OCW) 132
4.6.4 Modes of Operation 132
4.6.4.1 Fully Nested Mode 132
4.6.4.2 Special Fully Nested Mode 132
4.6.4.3 Automatic Rotation Mode (Equal Priority Devices) 133
4.6.4.4 Specific Rotation Mode (Specific Priority) 133
4.6.4.5 Poll Mode 133
4.6.4.6 Cascade Mode 133
4.6.4.7 Edge and Level Triggered Mode 134
4.6.4.8 End of Interrupt (EOI) Operations 134
4.6.4.9 Normal End of Interrupt 134

4.6.4.10 Automatic End of Interrupt Mode 134
4.6.5 Masking Interrupts 134
4.6.5.1 Masking on an Individual Interrupt Request 134
4.6.5.2 Special Mask Mode 135
4.6.6 Steering PCI Interrupts 135
4.7 Advanced Programmable Interrupt Controller (APIC) (B0:D31:F0) 135
4.7.1 Interrupt Handling 135
4.7.2 Interrupt Mapping 136
4.7.3 PCI Express* Message-Based Interrupts 136
4.7.4 IOxAPIC Address Remapping 137
4.8 Serial Interrupt (B0:D31:F0) 137
4.8.1 Start Frame 137
4.8.2 Data Frames 138
4.8.3 Stop Frame 138
4.8.4 Specific Interrupts Not Supported via SERIRQ 138
4.8.5 Data Frame Format 139
4.9 Real Time Clock (B0:D31:F0) 139
4.9.1 Update Cycles 140
4.9.2 Interrupts 140
4.9.3 Lockable RAM Ranges 140
4.9.4 Century Rollover 141
4.9.5 Clearing Battery-Backed RTC RAM 141
4.10 Processor Interface (B0:D31:F0) 142
4.10.1 Processor Interface Signals and VLW Messages 143
4.10.1.1 A20M# (Mask A20) / A20GATE 143
4.10.1.2 INIT (Initialization) 143
4.10.1.3 FERR# (Numeric Coprocessor Error) 143
4.10.1.4 NMI (Non-Maskable Interrupt) 143
4.10.1.5 Processor Power Good (CPUPWRGD) 144
4.10.2 Dual-Processor Issues 144
4.10.2.1 Usage Differences 144
4.10.3 Virtual Legacy Wire (VLW) Messages 144
4.11 Power Management (B0:D31:F0) 144
4.11.1 Features 144
4.11.2 System Power States 145
4.11.3 System Power Planes 146
4.11.4 SMI#/SCI Generation 147
4.11.4.1 PCI Express* SCI 149
4.11.4.2 PCI Express* Hot-Plug 149
4.11.5 C-States 150
4.11.6 Sleep States 150
4.11.6.1 Sleep State Overview 150
4.11.6.2 Initiating Sleep State 150
4.11.6.3 Exiting Sleep States 150
4.11.6.4 PCI Express* WAKE# Signal and PME Event Message 152
4.11.6.5 Sx-G3-Sx, Handling Power Failures 152
4.11.7 Event Input Signals and Their Usage 152
4.11.7.1 PWRBTN# (Power Button) 152
4.11.7.2 RI# (Ring Indicator) 154
4.11.7.3 SYS_RESET# Signal 154
4.11.7.4 THRMTRIP# Signal 154
4.11.8 ALT Access Mode 155
4.11.8.1 Write Only Registers with Read Paths in ALT Access Mode 155
4.11.8.2 PIC Reserved Bits 157
4.11.8.3 Read Only Registers with Write Paths in ALT Access Mode 158
4.11.9 System Power Supplies, Planes, and Signals 158

4.11.9.1 Power Plane Control with SLP_S3#, SLP_S4# and SLP_S5# 158
4.11.9.2 SLP_S4# and Suspend-To-RAM Sequencing 158
4.11.9.3 PWROK Signal 159
4.11.9.4 BATLOW# (Battery Low) 159
4.11.10 Legacy Power Management Theory of Operation 159
4.11.10.1 APM Power Management 159
4.11.11 Reset Behavior 159
4.12 System Management 161
4.12.1 SMBus Host Controller (B0:D31:F3) 161
4.12.1.1 Host Controller Interface 162
4.12.1.2 Command Protocols 163
4.12.1.3 Bus Arbitration 166
4.12.1.4 Bus Timing 166
4.12.1.5 Clock Stretching 166
4.12.1.6 Bus Time Out (Host as SMBus Master) 166
4.12.1.7 Interrupts / SMI# 167
4.12.1.8 SMBALERT# 168
4.12.1.9 SMBus CRC Generation and Checking 168
4.12.2 TCO Slave SMBus Interface 168
4.12.2.1 Format of Slave Write Cycle 169
4.12.2.2 Format of Read Command 170
4.12.2.3 Behavioral Notes 172
4.12.2.4 Slave Read of RTC Time Bytes 173
4.12.2.5 Format of Host Notify Command 173
4.12.2.6 TCO Slave Functions 174
4.12.2.7 Theory of Operation 174
4.12.3 EndPoint (EP) Slave SMBus 175
4.12.3.1 SMBus Supported Transactions 176
4.12.3.2 Addressing 177
4.12.3.3 SMBus Initiated Southbound Configuration Cycles 179
4.12.3.4 SMBus Error Handling 179
4.12.3.5 SMBus Interface Reset 179
4.12.3.6 Configuration and Memory Read Protocol 180
4.12.3.7 Configuration and Memory Write Protocol 183
4.12.4 GbE SMBus (Master/Slave) 185
4.12.5 SMLINK1 Interface 186
4.13 Serial I/O Unit and Watchdog Timer (SIW) (B0:D31:F0) 186
4.13.1 Overview 186
4.13.2 Features 186
4.13.3 Functional Description 187
4.13.3.1 Host Processor Interface (LPC) 187
4.13.3.2 LPC Interface 187
4.13.3.3 LPC Cycles 187
4.13.3.4 I/O Read and Write Cycles 187
4.13.3.5 Policy 188
4.13.3.6 LPC Transfers 188
4.13.4 LPC Logical Devices 4 and 5: Serial Ports (UART1 and UART2) 188
4.13.4.1 UART Feature List 189
4.13.4.2 UART Operational Description 190
4.13.4.3 Programmable Baud Rate Generator 191
4.13.4.4 FIFO Operation 191
4.13.4.5 FIFO Polled Mode Operation 192
4.13.5 LPC Logical Device 6: Watchdog Timer 193
4.13.5.1 Overview 193
4.13.5.2 Theory Of Operation 194
4.14 General Purpose I/O (B0:D31:F0) 194
4.14.1 Power Wells 195
4.14.2 SMI# SCI and NMI Routing 195

4.14.3 Triggering 195
4.14.4 GPIO Registers Lockdown 195
4.14.5 Serial POST Codes Over GPIO 196
4.14.5.1 Theory of Operation 196
4.14.5.2 Serial Message Format 197
4.15 SATA* Host Controller (B0:D31:F2 & B0:D31:F5) 198
4.15.1 SATA* Ports 4 and 5 Numbering 198
4.15.2 SATA* Feature Support 199
4.15.3 Theory of Operation 199
4.15.3.1 Standard ATA Emulation 199
4.15.3.2 48-Bit LBA Operation 199
4.15.4 SATA* Swap Bay Support 200
4.15.5 Hot Plug Operation 200
4.15.5.1 Low Power Device Presence Detection 200
4.15.6 Power Management Operation 200
4.15.6.1 Power State Mappings 200
4.15.6.2 Power State Transitions 201
4.15.6.3 SMI Trapping (APM) 202
4.15.7 SATALED# 202
4.15.8 AHCI Operation 202
4.15.9 SGPIO Signals 202
4.15.9.1 Mechanism 203
4.15.9.2 Message Format 204
4.15.9.3 LED Message Type 204
4.15.9.4 SGPIO Waveform 206
4.16 High Precision Event Timers 206
4.16.1 Timer Accuracy 207
4.16.2 Interrupt Mapping 207
4.16.3 Periodic Versus Non-Periodic Modes 208
4.16.4 Enabling the Timers 208
4.16.5 Interrupt Levels 208
4.16.6 Handling Interrupts 209
4.16.7 Issues Related to 64-Bit Timers with 32-Bit Processors 209
4.17 USB* EHCI Host Controllers (B0:D29:F0) 209
4.17.1 EHC Initialization 210
4.17.1.1 BIOS Initialization 210
4.17.1.2 Driver Initialization 210
4.17.1.3 EHC Resets 210
4.17.2 Data Structures in Main Memory 210
4.17.3 USB* 2.0 Enhanced Host Controller DMA 211
4.17.4 Data Encoding and Bit Stuffing 211
4.17.5 Packet Formats 211
4.17.6 USB* 2.0 Interrupts and Error Conditions 211
4.17.6.1 Aborts on USB* 2.0-Initiated Memory Reads 212
4.17.7 USB* 2.0 Power Management 212
4.17.7.1 Pause Feature 212
4.17.7.2 Suspend Feature 212
4.17.7.3 ACPI Device States 212
4.17.7.4 ACPI System States 213
4.17.8 USB* 2.0 Legacy Keyboard Operation 213
4.17.9 USB* 2.0 Based Debug Port 213
4.17.9.1 Theory of Operation 214
4.17.10 EHCI Caching 218
4.17.11 USB* Pre-Fetch Based Pause 218
4.17.12 USB* Overcurrent Protection 218
4.18 Integrated USB* 2.0 Rate Matching Hub 219

4.18.1 Architecture 219
4.19 Thermal Management 219
4.19.1 Modes of Operation 219
4.19.2 Thermal Reporting Over System Management Link 1 Interface (SMLink1) 220
4.20 WatchDog Timer (WDT) Host Controller (B0:D31:F7) 221
4.20.1 Theory Of Operation 221
4.20.2 Watchdog Timer Behavior 221
4.20.3 Pending Bit Array 221
4.20.4 MSI Interrupt Formation 222
4.20.5 Usage Models 222
4.21 Serial Peripheral Interface (SPI) (B0:D31:F0) 222
4.21.1 Descriptor Mode 223
4.21.2 SPI Flash Regions 223
4.21.2.1 Device Partitioning 224
4.21.3 Flash Descriptor 224
4.21.3.1 Descriptor Master Region 226
4.21.4 Flash Access 226
4.21.4.1 Direct Access Security 226
4.21.4.2 Register Access Security 227
4.21.5 Serial Flash Device Compatibility Requirements 227
4.21.5.1 SPI Based BIOS Requirements 227
4.21.5.2 Intel® Management Engine Firmware SPI Flash Requirements 228
4.21.5.3 Hardware Sequencing Requirements 228
4.21.6 Multiple Page Write Usage Model 229
4.21.6.1 Soft Flash Protection 229
4.21.6.2 BIOS Range Write Protection 230
4.21.6.3 SMI# Based Global Write Protection 230
4.21.7 Flash Device Configurations 230
4.21.8 SPI Flash Device Recommended Pinout 230
4.21.9 Serial Flash Device Package 231
4.21.9.1 Common Footprint Usage Model 231
4.21.9.2 Serial Flash Device Package Recommendations 231
4.22 Feature Capability Mechanism 232

5.0 Register and Memory Mappings 233
5.1 I/O Map 233
5.1.1 Fixed I/O Address Ranges 233
5.1.2 Variable I/O Decode Ranges 235
5.2 Memory Map 236
5.3 Boot-Block Update Scheme 237

6.0 LPC Interface Bridge Registers (B0:D31:F0) 239
6.1 Overview 239
6.1.1 PCI Configuration Registers (LPC I/F—B0:D31:F0) 239
6.1.1.1 Offset 00h: VID—Vendor Identification Register 241
6.1.1.2 Offset 02h: DID—Device Identification Register 241
6.1.1.3 Offset 04h: ID—PCICMD—PCI COMMAND Register 242
6.1.1.4 Offset 06h: ID—PCISTS—PCI Status Register 243
6.1.1.5 Offset 08h: RID—Revision Identification Register 244
6.1.1.6 Offset 09h: PI—Programming Interface Register 244
6.1.1.7 Offset 0Ah: SCC—Sub Class Code Register 244
6.1.1.8 Offset 0Bh: BCC—Base Class Code Register 245
6.1.1.9 Offset 0Dh: PLT—Primary Latency Timer Register 245
6.1.1.10 Offset 0Eh: HEADTYP—Header Type Register 245
6.1.1.11 Offset 2Ch: SS—Sub System Identifiers Register 246
6.1.1.12 Offset 40h: PMBASE—ACPI Base Address Register 246
6.1.1.13 Offset 44h: ACPI_CNTL—ACPI Control Register 247

6.1.1.14 Offset 48h: GPIOBASE—GPIO Base Address Register 248
6.1.1.15 Offset 4Ch: GC—GPIO Control Register 249
6.1.1.16 Offset 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control Register 250
6.1.1.17 Offset 64h: SIRQ_CNTL—Serial IRQ Control Register 251
6.1.1.18 Offset 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control Register 252
6.1.1.19 Offset 6Ch: LPC_IBDF—IOxAPIC Bus:Device:Function 253
6.1.1.20 Offset 70h: LPC_HnBDF—HPET n Bus:Device:Function 254
6.1.1.21 Offset 80h: LPC_I/O_DEC—I/O Decode Ranges Register 255
6.1.1.22 Offset 82h: LPC_EN—LPC I/F Enables Register 256
6.1.1.23 Offset 84h: GEN1_DEC—LPC I/F Generic Decode Range 1 Register 258
6.1.1.24 Offset 88h: GEN2_DEC—LPC I/F Generic Decode Range 2 Register 259
6.1.1.25 Offset 8Ch: GEN3_DEC—LPC I/F Generic Decode Range 3 Register 260
6.1.1.26 Offset 90h: GEN4_DEC—LPC I/F Generic Decode Range 4 Register 261
6.1.1.27 Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control 262
6.1.1.28 Offset 98h: LGMR—LPC I/F Generic Memory Range 264
6.1.1.29 Offset DCh: BIOS_CNTL—BIOS Control Register 265
6.1.1.31 Offset E2h: FDLEN—Feature Detection Capability Length 266
6.1.1.32 Offset E3h: FDVER—Feature Detection Version 267
6.1.1.33 Offset E4h: FDVCT—Feature Vector 267
6.1.1.34 Offset F0h: RCBA—Root Base Address Register 268
6.1.2 DMA I/O Registers 268
6.1.2.1 Offset 00h: DMABASE_CA—DMA Base and Current Address Registers 269
6.1.2.3 Offset 87h: DMAMEM_LP—DMA Memory Low Page Registers (LPC I/F—B0:D31:F0) 271
6.1.2.4 Offset 08h: DMACMD—DMA Command Register 272
6.1.2.6 Offset 0Ah: DMA_WRSMSK—DMA Write Single Mask Register 274
6.1.2.7 Offset 0Bh: DMACH_MODE—DMA Channel Mode Register 275
6.1.2.8 Offset 0Ch: DMA Clear Byte Pointer Register 276
6.1.2.9 Offset 0Dh: DMA Master Clear Register 276
6.1.2.10 Offset 0Eh: DMA_CLMSK—DMA Clear Mask Register 276
6.1.2.11 Offset 0Fh: DMA_WRMSK—DMA Write All Mask Register 277
6.2 Timer I/O Registers (LPC I/F—B0:D31:F0) 277
6.2.1 Timer I/O Registers 277
6.2.1.1 Offset 43h: TCW—Timer Control Word Register 278
6.2.1.2 Offset 40h: SBYTE_FMT—Interval Timer Status Byte Format Register 280
6.2.1.3 Offset 40h: Counter Access Ports Register 281
6.3 8259 Interrupt Controller (PIC) Registers 282
6.3.1 Interrupt Controller I/O MAP 282
6.3.1.1 Offset 20h: Master PIC ICW1—Master Initialization Command Word 1 Register 283
6.3.1.2 Offset 21h: Master PIC ICW2—Master Initialization Command Word 2 Register 284
6.3.1.3 Offset 21h: Master PIC ICW3—Master Initialization Command Word 3 Register 285
6.3.1.4 Offset A1h: Slave PIC ICW3—Slave Initialization Command Word 3 Register 285
6.3.1.5 Offset 021h: Master PIC ICW4—Master Initialization Command Word 4 Register 286
6.3.1.6 Offset 021h: Master PIC OCW1—Master Operational Control Word 1 (Interrupt Mask) Register 287
6.3.1.7 Offset 020h: Master PIC OCW2—Master Operational Control Word 2 Register 288
6.3.1.8 Offset 020h: Master PIC OCW3—Master Operational Control Word 3 Register 289
6.3.1.9 Offset 4D0h: Master PIC ELCR1—Master Controller Edge/Level Triggered Register 290
6.3.1.10 Offset 4D1h: Slave PIC ELCR2—Slave Controller Edge/Level Triggered Register 291
6.4 Advanced Programmable Interrupt Controller (APIC) 292

6.4.1 APIC Register Map 292
6.4.2 APIC Direct Registers 292
6.4.2.1 IND—Index Register 292
6.4.2.2 DAT—Data Register 293
6.4.2.3 EOIR—EOI Register 293
6.4.3 APIC Indirect Registers 294
6.4.3.1 Offset 00h: ID—Identification Register 295
6.4.3.2 Offset 01h: VER—Version Register 296
6.4.3.3 Offset 10h: REDIR_TBL0—Redirection Table 0 296
6.5 Real Time Clock Registers 299
6.5.1 I/O Register Address Map 299
6.5.2 Indexed Registers 300
6.5.2.1 Offset 0Ah: RTC_REGA—Register A 301
6.5.2.2 Offset 0Bh: RTC_REGB—Register B (General Configuration) 302
6.5.2.3 Offset OCh: RTC_REGC—Register C (Flag Register) 303
6.5.2.4 Offset ODh: D—Register D (Flag Register) 304
6.6 Processor Interface Registers (LPC I/F—B0:D31:F0) 305
6.6.1 Processor Interface PCI Registers Address Map 305
6.6.1.1 Offset 61h: NMI_SC—NMI Status and Control Register 306
6.6.1.2 Offset 70h: NMI_EN—NMI Enable (and Real Time Clock Index) Register . 307
6.6.1.3 Offset 92h: PORT92—Fast A20 and Init Register 307
6.6.1.4 Offset F0h: COPROC_ERR—Coprocessor Error Register 308
6.6.1.5 Offset CF9h: RST_CNT—Reset Control Register 308
6.7 Power Management Registers (PM—B0:D31:F0) 310
6.7.1 Power Management PCI Configuration Registers 310
6.7.1.1 Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register 311
6.7.1.2 Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register 313
6.7.1.3 Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register 315
6.7.1.4 Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register 318
6.7.1.5 Offset A9h: Chipset Initialization Register 4 318
6.7.1.6 Offset ABh: BM_BREAK_EN Register 319
6.7.1.7 Offset ACh: PMIR—Power Management Initialization Register 319
6.7.1.8 Offset B8h: GPIO_ROUT—GPIO Routing Control Register 320
6.7.2 APM I/O Decode 320
6.7.2.1 Offset B2h: APM_CNT—Advanced Power Management Control Port Register 321
6.7.2.2 Offset B3h: APM_STS—Advanced Power Management Status Port Register 321
6.7.3 Power Management I/O Registers 322
6.7.3.1 Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register .. 323
6.7.3.2 Offset PMBASE + 02h: PM1_EN—Power Management 1 Enable Register .. 326
6.7.3.3 Offset PMBASE + 04h: PM1_CNT—Power Management 1 Control 327
6.7.3.4 Offset PMBASE + 08h: PM1_TMR—Power Management 1 Timer Register . 328
6.7.3.5 Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register 329
6.7.3.6 Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register 331
6.7.3.7 Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register 333
6.7.3.8 Offset PMBASE + 34h: SMI_STS—SMI Status Register 335
6.7.3.9 Offset PMBASE +38h: ALT_GP_SMI_EN—Alternate GPI SMI Enable Register 338
6.7.3.10 Offset PMBASE +3Ah: ALT_GP_SMI_STS—Alternate GPI SMI Status Register 338
6.7.3.11 Offset PMBASE +3Ch: UPRWC—USB Per-Port Registers Write Control 339
6.7.3.12 Offset PMBASE +42h: GPE_CNTL—General Purpose Control Register 340
6.7.3.13 Offset PMBASE +44h: DEVACT_STS—Device Activity Status Register 341
6.7.3.14 Offset PMBASE +50h: PM2_CNT—Power Management 2 Control 342

6.8 System Management TCO Registers (B0:D31:F0) 342
6.8.1 TCO Register I/O Map 342
6.8.1.1 Offset TCOBASE +00h: TCO_RLD—TCO Timer Reload and Current Value Register 343
6.8.1.2 Offset TCOBASE +02h: TCO_DAT_IN—TCO Data In Register 343
6.8.1.3 Offset TCOBASE +03h: TCO_DAT_OUT—TCO Data Out Register 344
6.8.1.4 Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register 344
6.8.1.5 Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register 346
6.8.1.6 Offset TCOBASE +08h: TCO1_CNT—TCO1 Control Register 348
6.8.1.7 Offset TCOBASE +0Ah: TCO2_CNT—TCO2 Control Register 349
6.8.1.8 Offset TCOBASE +0Ch and Offset TCOBASE +0Dh: TCO_MESSAGE1 and TCO_MESSAGE2 Registers 350
6.8.1.9 Offset TCOBASE + 0Eh: TCO_WDCNT—TCO Watchdog Control Register . 350
6.8.1.10 Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register 351
6.9 General Purpose I/O Registers (B0:D31:F0) 352
6.9.1 General Purpose I/O Signals 352
6.9.1.1 Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register 353
6.9.1.2 Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register 354
6.9.1.3 Offset GPIOBASE + 0Ch:GP_LVL—GPIO Level for Input or Output Register 354
6.9.1.4 Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register 355
6.9.1.5 Offset GPIOBASE + 1Ch: GP_SER_BLINK—GP Serial Blink Data 356
6.9.1.6 Offset GPIOBASE + 20h: GP_SB_CMDSTS—GP Serial Blink Command Status 357
6.9.1.7 Offset GPIOBASE + 24h: GP_SB_DATA—GP Serial Blink Data 358
6.9.1.8 Offset GPIOBASE + 28h: GPI_NMI_EN—GPI NMI Enable 358
6.9.1.9 Offset GPIOBASE + 2Ah: GPI_NMI_STS—GPI NMI Status 359
6.9.1.10 Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register 359
6.9.1.11 Offset GPIOBASE + 30h: GPIO_USE_SEL2—GPIO Use Select 2 Register . 360
6.9.1.12 Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register 361
6.9.1.13 Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register 361
6.9.1.14 Offset GPIOBASE + 40h: GPIO_USE_SEL3—GPIO Use Select 3 Register . 362
6.9.1.15 Offset GPIOBASE + 44h: GP_IO_SEL3—GPIO Input/Output Select 3 Register 363
6.9.1.16 Offset GPIOBASE + 48h: GP_LVL3—GPIO Level for Input or Output 3 Register 364
6.9.1.17 Offset GPIOBASE + 60h: GP_RST_SEL1—GPIO Reset Select 365
6.9.1.18 Offset GPIOBASE + 64h: GP_RST_SEL2—GPIO Reset Select 366
6.9.1.19 Offset GPIOBASE + 68h: GP_RST_SEL3—GPIO Reset Select 366

7.0 Chipset Configuration Registers (B0:D31:F0) 367
7.1 Chipset Configuration Registers 367
7.1.1 Chipset Configuration Register Memory Map (Memory Space) 367
7.1.1.1 Offset 0014h: V0CTL—Virtual Channel 0 Resource Control Register 370
7.1.1.2 Offset 001Ah: V0STS—Virtual Channel 0 Resource Status Register 371
7.1.1.3 Offset 001Ch: V1CAP—Virtual Channel 1 Resource Capability Register ... 371
7.1.1.4 Offset 0020h: V1CTL—Virtual Channel 1 Resource Control Register 372
7.1.1.5 Offset 0026h: V1STS—Virtual Channel 1 Resource Status Register 372
7.1.1.6 Offset 0050h: CIR0—Chipset Initialization Register 0 373
7.1.1.7 Offset 0088h: CIR1—Chipset Initialization Register 1 373
7.1.1.8 Offset 00ACh: REC—Root Error Command Register 374
7.1.1.9 Offset 01A0h: ILCL—Internal Link Capabilities List Register 374
7.1.1.10 Offset 01A4h: LCAP—Link Capabilities Register 375
7.1.1.11 Offset 01A8h: LCTL—Link Control Register 375
7.1.1.12 Offset 01AAh: LSTS—Link Status Register 376
7.1.1.13 Offset 0220h: BCR—Backbone Configuration Register 376

7.1.1.14 Offset 0224h: RPC—Root Port Configuration Register 377
7.1.1.15 Offset 0234h: DMIC—DMI Control Register 378
7.1.1.16 Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports 378
7.1.1.17 Offset 0290h: Reserved Register 379
7.1.1.18 Offset 1D40H: CIR5—Chipset Initialization Register 5 380
7.1.1.19 Offset 1E00h: TRSR—Trap Status Register 380
7.1.1.20 Offset 1E10h: TRCR—Trapped Cycle Register 381
7.1.1.21 Offset 1E18h: TWDR—Trapped Write Data Register 381
7.1.1.22 Offset 1E80h: IOTRn—I/O Trap Register (0-3) 382
7.1.1.23 Offset 2010h: DMC—DMI Miscellaneous Control Register 383
7.1.1.24 Offset 2024h: CIR6—Chipset Initialization Register 6 383
7.1.1.25 Offset 2324h: DMC2—DMI Miscellaneous Control Register 2 384
7.1.1.26 Offset 60h: SBI Unified AFE Address Register 384
7.1.1.27 Offset 64h: SSBI Unified AFE Data Register 385
7.1.1.28 Offset 68h: SBI Unified AFE Status Register 386
7.1.1.29 Offset 3000h: TCTL—TCO Configuration Register 387
7.1.1.30 Offset 3100h: D31IP—Device 31 Interrupt Pin Register 388
7.1.1.31 Offset 3108h: D29IP—Device 29 Interrupt Pin Register 389
7.1.1.32 Offset 310Ch: D28IP—Device 28 Interrupt Pin Register 390
7.1.1.33 Offset 3124h: D22IP—Device 22 Interrupt Pin Register 391
7.1.1.34 Offset 3140h: D31IR—Device 31 Interrupt Route Register 392
7.1.1.35 Offset 3144h: D29IR—Device 29 Interrupt Route Register 393
7.1.1.36 Offset 3146h: D28IR—Device 28 Interrupt Route Register 394
7.1.1.37 Offset 315Ch: D22IR—Device 22 Interrupt Route Register 395
7.1.1.38 Offset 31FEh: OIC—Other Interrupt Control Register 396
7.1.1.39 Offset 3310h: PRSTS—Power and Reset Status 397
7.1.1.40 Offset 3314h: CIR7—Chipset Initialization Register 7 398
7.1.1.41 Offset 3324h: CIR8—Chipset Initialization Register 8 398
7.1.1.42 Offset 3330h: CIR9—Chipset Initialization Register 9 398
7.1.1.43 Offset 3340h: CIR10—Chipset Initialization Register 10 399
7.1.1.44 Offset 3350h: CIR13—Chipset Initialization Register 13 399
7.1.1.45 Offset 3368h: CIR14—Chipset Initialization Register 14 399
7.1.1.46 Offset 3378h: CIR15—Chipset Initialization Register 15 400
7.1.1.47 Offset 3388h: CIR16—Chipset Initialization Register 16 400
7.1.1.48 Offset 33A0h: CIR17—Chipset Initialization Register 17 400
7.1.1.49 Offset 33A8h: CIR18—Chipset Initialization Register 18 401
7.1.1.50 Offset 33C0h: CIR19—Chipset Initialization Register 19 401
7.1.1.51 Offset 33CCh: CIR20—Chipset Initialization Register 20 401
7.1.1.52 Offset 33D0h: CIR21—Chipset Initialization Register 21 402
7.1.1.53 Offset 33D4h: CIR22—Chipset Initialization Register 22 402
7.1.1.54 Offset 3400h: RC—RTC Configuration Register 403
7.1.1.55 Offset 3404h: HPTC—High Precision Timer Configuration Register 404
7.1.1.56 Offset 3410h: GCS—General Control and Status Register 405
7.1.1.57 Offset 3414h: BUC—Backed Up Control Register 407
7.1.1.58 Offset 3418h: FD—Function Disable Register 408
7.1.1.59 Offset 341Ch: CG—Clock Gating 410
7.1.1.60 Offset 3420h: FDSW—Function Disable SUS Well 411
7.1.1.61 Offset 3428h: FD2—Function Disable 2 411
7.1.1.62 Offset 3500h: USBIR[0:5]—USB Initialization Register [0-5] 412
7.1.1.63 Offset 3564h: USBIRC—USB Initialization Register C 413
7.1.1.64 Offset 3570h: USBIRA—USB Initialization Register A 413
7.1.1.65 Offset 357Ch: USBIRB—USB Initialization Register B 414
7.1.1.66 Offset 3590h: MISCCTL—Miscellaneous Control Register 415
7.1.1.67 Offset 359Ch: PDO—USB Port Disable Override 416
7.1.1.68 Offset 35A0h: USBOCM1—Overcurrent MAP Register 1 417
7.1.1.69 Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register ... 418

8.0 SATA* Controller Registers (B0:D31:F2) 420
8.1 PCI Configuration Registers 420

8.1.1 SATA* Controller PCI Register Address Map 420
8.1.1.1 Offset 00h: Vendor Identification Register 422
8.1.1.2 Offset 02h: Device Identification Register 422
8.1.1.3 Offset 04h: PCI Command Register 423
8.1.1.4 Offset 06h: PCI Status Register 424
8.1.1.5 Offset 08h: Revision Identification Register 425
8.1.2 Programming Interface Register 425
8.1.2.1 Offset 0Ah: When Sub Class Code Register = 01h 425
8.1.2.2 Offset 0Ah: When Sub Class Code Register = 06h 426
8.1.2.3 Offset 0Ah: Sub Class Code Register 427
8.1.2.4 Offset 0Bh: Base Class Code Register 427
8.1.2.5 Offset 0Dh: Primary Master Latency Timer Register 427
8.1.2.6 Offset 0Eh: Header Type 428
8.1.2.7 Offset 10h: Primary Command Block Base Address Register 428
8.1.2.8 Offset 14h: Primary Control Block Base Address Register 429
8.1.2.9 Offset 18h: Secondary Command Block Base Address Register 429
8.1.2.10 Offset 1Ch: Secondary Control Block Base Address Register 430
8.1.2.11 Offset 20h: Legacy Bus Master Base Address Register 430
8.1.3 AHCI Base Address Register/Serial ATA Index Data Pair Base Address 431
8.1.3.1 Offset 24h: When SCC is Not 01h 431
8.1.3.2 Offset 24h: When SCC is 01h 432
8.1.3.3 Offset 2Ch: Subsystem Vendor Identification Register 432
8.1.3.4 Offset 2Eh: Subsystem Identification Register 433
8.1.3.5 Offset 34h: Capabilities Pointer Register 433
8.1.3.6 Offset 3Ch: Interrupt Line Register 433
8.1.3.7 Offset 3Dh: Interrupt Pin Register 434
8.1.3.8 Offset 40h: IDE Timing Register 434
8.1.3.9 Offset 48h: Synchronous DMA Control Register 435
8.1.3.10 Offset 4Ah: Synchronous DMA Timing Register 435
8.1.3.11 Offset 54h: IDE I/O Configuration Register 436
8.1.4 PCI Power Management Capabilities 436
8.1.4.1 Offset 70h: PCI Power Management Capability Identification Register 436
8.1.4.2 Offset 72h: PCI Power Management Capabilities Register 437
8.1.4.3 Offset 74h: PCI Power Management Control and Status Register 438
8.1.5 Message Signaled Interrupt Capability 439
8.1.5.1 Offset 80h: Message Signaled Interrupt Capability Identification 439
8.1.5.2 Offset 82h: Message Signaled Interrupt Message Control 440
8.1.5.3 Offset 84h: Message Signaled Interrupt Message Address 441
8.1.5.5 Offset 90h: MAP—Address Map Register 442
8.1.5.6 Offset 92h: PCS - Port Control and Status Register 443
8.1.5.7 Offset 94h: SCLKCG—SATA Clock Gating Control Register 445
8.1.5.8 Offset 9Ch: SCLKGC—SATA Clock General Configuration Register 446
8.1.5.9 Offset A0h: SIRI—SATA Indexed Registers Index 447
8.1.5.10 Offset A4h: STRD—SATA Indexed Register Data 447
8.1.5.11 Offset A8h: SATACR0—SATA Capability Register 0 448
8.1.5.12 Offset ACh: SATACR1—SATA Capability Register 1 449
8.1.5.13 Offset B0h: FLRCID—FLR Capability ID 450
8.1.5.14 Offset B2h: FLRCLV—FLR Capability Length and Version 450
8.1.5.15 Offset B4h: FLRC—FLR Control 451
8.1.5.16 Offset C0h: ATC—APM Trapping Control Register 452
8.1.5.17 Offset C4h: ATS—APM Trapping Status Register 452
8.1.5.18 Offset D0h: SP—Scratch Pad Register 453
8.1.5.19 Offset E0h: BFCS—BIST FIS Control/Status Register 453
8.1.5.20 Offset E4h: BFTD1—BIST FIS Transmit Data1 Register 455
8.1.5.21 Offset E8h: BFTD2—BIST FIS Transmit Data2 Register 455
8.1.6 SATA* Indexed Registers 456
8.1.6.1 Offset 18h: SATA* Indexed Registers Index (SATA* Initialization Register) 457

8.1.6.2 Offset 1Ch: SATA* Indexed Registers Index (SATA* Test Mode Enable Register) 457
8.1.6.3 Offset 28h: SATA* Indexed Registers Index (SATA* Initialization Register) 458
8.1.6.4 Offset 3Eh: SATA* Indexed Registers Index (SATA* Initialization Register) 458
8.1.6.5 Offset 54h: SATA* Indexed Registers Index (SATA* Initialization Register) 459
8.1.6.6 Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) 459
8.1.6.7 Offset 68h: SATA* Indexed Registers Index (SATA* Initialization Register) 460
8.1.6.8 Offset 78h: SATA* Indexed Registers Index (SATA* Initialization Register) 460
8.1.6.9 Offset 84h: SATA* Indexed Registers Index (SATA* Initialization Register) 461
8.1.6.10 Offset 88h: SATA* Indexed Registers Index (SATA* Initialization Register) 461
8.1.6.11 Offset 8Ch: SATA* Indexed Registers Index (SATA* Initialization Register) 462
8.1.6.12 Offset 94h: SATA* Indexed Registers Index (SATA* Initialization Register) 462
8.1.6.13 Offset A0h: SATA* Indexed Registers Index (SATA* Initialization Register) 463
8.1.6.14 Offset A8h: SATA* Indexed Registers Index (SATA* Initialization Register) 463
8.1.6.15 Offset C4h: SATA* Indexed Registers Index (SATA* Initialization Register) 464
8.1.6.16 Offset C8h: SATA* Indexed Registers Index (SATA* Initialization Register) 464
8.2 Bus Master IDE I/O Registers (B0:D31:F2) 465
8.2.1 Bus Master IDE I/O Register Address Map 465
8.2.1.1 Offset 00h: Bus Master IDE Command Register Primary 466
8.2.1.2 Offset 02h: Bus Master IDE Status Register Primary 467
8.2.1.3 Offset 04h: Bus Master IDE Descriptor Table Pointer Register Primary 468
8.2.1.4 Offset 08h: Bus Master IDE Command Register Secondary 469
8.2.1.5 Offset 0Ah: Bus Master IDE Status Register Secondary 470
8.2.1.6 Offset 0Ch: Bus Master IDE Descriptor Table Pointer Register Secondary 471
8.2.1.7 Offset 10h: AHCI Index Register 471
8.2.1.8 Offset 14h: AHCI Index Data Register 472
8.3 Serial ATA Index/Data Pair Superset Registers 473
8.3.1 Superset Registers 473
8.3.1.1 Offset SIDPBA + 00h: Serial ATA Index 473
8.3.1.2 Offset 04h: Serial ATA Data 474
8.4 AHCI Registers 474
8.4.1 AHCI Generic Host Control Registers 475
8.4.1.1 Offset 00h: Host Capabilities Register 475
8.4.1.2 Offset 04h: Global PCH Control Register 478
8.4.1.3 Offset 08h: Interrupt Status Register 479
8.4.1.4 Offset 0Ch: Ports Implemented Register 480
8.4.1.5 Offset 10h: Serial AHCI Version 480
8.4.1.6 Offset 14h: Serial Command Completion Coalescing Control Register 481
8.4.1.7 Offset 18h: Serial Command Completion Coalescing Ports Register 482
8.4.1.8 Offset 1Ch: Serial Enclosure Management Location Register 482
8.4.1.9 Offset 20h: Serial Enclosure Management Control Register 483
8.4.1.10 Offset 24h: Serial Extended Host Capabilities 484
8.4.1.11 Offset 6Ch: Serial NVMHCI Ports Implemented 485
8.4.1.12 Offset 70h: Serial AHCI Version 485
8.4.1.13 Offset A0h: Serial Vendor Specific 486

8.4.2 Port Registers 487
8.4.2.1 Offset 300h: Port [4:5] Command List Base Address Register 488
8.4.2.2 Offset 304h: Port [4:5] Command List Base Address Upper 32-Bits Register 488
8.4.2.3 Offset 308h: Port [4:5] FIS Base Address Register 489
8.4.2.4 Offset 30Ch: Port [4:5] FIS Base Address Upper 32-Bits Register 489
8.4.2.5 Offset 310h: Port [4:5] Interrupt Status Register 490
8.4.2.6 Offset 314h: Port [4:5] Interrupt Enable Register 492
8.4.2.7 Offset 318h: Port [4:5] Command Register 494
8.4.2.8 Offset 320h: Port [4:5] Task File Data Register 497
8.4.2.9 Offset 324h: Port [4:5] Signature Register 498
8.4.2.10 Offset 328h: Port [4:5] Serial ATA Status Register 499
8.4.2.11 Offset 32Ch: Port [4:5] Serial ATA Control Register 501
8.4.2.12 Offset 330h: Port [4:5] Serial ATA Error Register 503
8.4.2.13 Offset 334h: Port [4:5] Serial ATA Active 504
8.4.2.14 Offset 338h: Port [4:5] Command Issue Register 505

9.0 SATA* Controller Registers (B0:D31:F5) 506
9.1 PCI Configuration Registers (SATA–B0:D31:F5) 506
9.1.1 SATA* Controller PCI Register Address Map 506
9.1.1.1 Offset 00h: Vendor Identification Register 507
9.1.1.2 Offset 02h: Device Identification Register 508
9.1.1.3 Offset 04h: PCI Command Register 508
9.1.1.4 Offset 06h: PCI Status Register 509
9.1.1.5 Offset 08h: RID—Revision Identification Register 510
9.1.1.6 Offset 09h: Programming Interface Register 511
9.1.1.7 Offset 0Ah: Sub Class Code Register 511
9.1.1.8 Offset 0Bh: BCC—Base Class Code Register 512
9.1.1.9 Offset 0Dh: Primary Master Latency Timer Register 512
9.1.1.10 Offset 10h: Primary Command Block Base Address Register 513
9.1.1.11 Offset 14h: Primary Control Block Base Address Register 514
9.1.1.12 Offset 18h: SCMD-Secondary Command Block Base Address Register 514
9.1.1.13 Offset 1Ch: SCNL- Secondary Control Block Base Address Register 515
9.1.1.14 Offset 20h: Legacy Bus Master Base Address Register 515
9.1.1.15 Offset 24h: SATA Index/Data Pair Base Address Register 516
9.1.1.16 Offset 2Ch: Subsystem Vendor Identification Register 516
9.1.1.17 Offset 2Eh: Subsystem Identification Register 517
9.1.1.18 Offset 34h: Capabilities Pointer Register 517
9.1.1.19 Offset 3Ch: Interrupt Line Register 517
9.1.1.21 Offset 40h: IDE Timing Register 518
9.1.1.22 Offset 48h: Synchronous DMA Control Register 519
9.1.1.23 Offset 4Ah: Synchronous DMA Timing Register 519
9.1.1.24 Offset 54h: IDE I/O Configuration Register 520
9.1.1.25 Offset 70h: PCI Power Management Capability Identification Register 521
9.1.1.26 Offset 72h: PC—PCI Power Management Capabilities Register 521
9.1.1.27 Offset 74h: PCI Power Management Control and Status Register 522
9.1.1.28 Offset 90h: MAP—Address Map Register 523
9.1.1.29 Offset 92h: Port Control and Status Register 524
9.1.1.30 Offset A8h: SATA Capability Register 0 525
9.1.1.31 Offset ACh: SATA* Capability Register 1 525
9.1.1.32 Offset B0h: FLR Capability ID 526
9.1.1.33 Offset B2h: FLR Capability Length and Value 526
9.1.1.34 Offset B4h: FLR Control 527
9.1.1.35 Offset C0h: APM Trapping Control Register 528
9.1.1.36 Offset C4h: APM Trapping Control Register 528
9.1.2 Bus Master IDE I/O Registers (B0:D31:F5) 529
9.1.2.1 Offset 00h: Bus Master IDE Command Register 529
9.1.2.2 Offset 02h: Bus Master IDE Status Register 531
9.1.2.3 Offset 04h: Bus Master IDE Descriptor Table Pointer Register 532
9.1.3 Serial ATA Index/Data Pair Superset Registers 532

9.1.3.1 Offset 00h: SINDX—SATA* Index Register 533
9.1.3.2 Offset 04h: SDATA—SATA* Index Data Register 533
9.1.3.3 Offset 04h: PxSSTS—Serial ATA Status Register 534
9.1.3.4 Offset 04h: PxSCTL—Serial ATA Control Register 536
9.1.3.5 Offset 04h: PxSERR—Serial ATA Error Register 538

10.0 EHCI Controller Registers (B0:D29:F0) 540
10.1 USB* EHCI Configuration Registers (USB EHCI—B0:D29:F0) 540
10.1.1 USB* EHCI PCI Register Address Map 540
10.1.1.1 Offset 00h: Vendor Identification Register 543
10.1.1.2 Offset 02h: Device Identification Register 543
10.1.1.3 Offset 04h: PCI Command Register 544
10.1.1.4 Offset 06h: PCI Status Register 545
10.1.1.5 Offset 08h: RID—Revision Identification Register 546
10.1.1.6 Offset 09h: Programming Interface Register 547
10.1.1.7 Offset 0Ah: Sub Class Code Register 547
10.1.1.8 Offset 0Bh: BCC—Base Class Code Register 547
10.1.1.9 Offset 0Dh: Primary Master Latency Timer Register 548
10.1.1.10 Offset 0Eh: Header Type Register 549
10.1.1.11 Offset 10h: Memory Base Address Register 549
10.1.1.12 Offset 2Ch: Subsystem Vendor ID Register 550
10.1.1.13 Offset 2Eh: Subsystem ID Register 550
10.1.1.14 Offset 34h: Capabilities Pointer Register 551
10.1.1.15 Offset 3Ch: Interrupt Line Register 551
10.1.1.16 Offset 3Dh: Interrupt Pin Register 551
10.1.1.17 Offset 50h: PCI Power Management Capability ID Register 552
10.1.1.18 Offset 51h: Next Item Pointer #1 Register 552
10.1.1.19 Offset 52h: Power Management Capabilities Register 553
10.1.1.20 Offset 54h: Power Management Control/Status Register 554
10.1.1.21 Offset 58h: Debug Port Capability ID Register 555
10.1.1.22 Offset 59h: Next Item Pointer #2 Register 555
10.1.1.23 Offset 5Ah: Debug Port Base Offset Register 555
10.1.1.24 Offset 60h: USB* Release Number Register 556
10.1.1.25 Offset 61h: Frame Length Adjustment Register 556
10.1.1.26 Offset 62h: Port Wake Capability Register 558
10.1.1.27 Offset 68h: Legacy Support Extended Capability Register 559
10.1.1.28 Offset 6Ch: Legacy Support Extended Control/Status Register 560
10.1.1.29 Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register 563
10.1.1.30 Offset 80h: ACCESS_CNTL—Access Control Register 565
10.1.1.31 Offset 84h: EHCIIR1—EHCI Initialization Register 1 565
10.1.1.32 Offset 98h: FLR_CID—Function Level Reset Capability ID 566
10.1.1.33 Offset 99h: FLR_NEXT—Function Level Reset Next Capability Pointer 566
10.1.1.34 Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version 567
10.1.1.35 Offset 9Ch: FLR_CTRL—Function Level Reset Control Register 568
10.1.1.36 Offset 9Dh: FLR_STS—Function Level Reset Status Register 568
10.2 Memory-Mapped I/O Registers 569
10.2.1 Host Controller Capability Registers 569
10.2.1.1 Offset 00h: CAPLENGTH—Capability Registers Length Register 570
10.2.1.2 Offset 02h: HCIVERSION—Host Controller Interface Version Number Register 570
10.2.1.3 Offset 04h: HCSPARAMS—Host Controller Structural Parameters 571
10.2.1.4 Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB* EHCI—B0:D29:F0) 572
10.2.2 Host Controller Operational Registers 573
10.2.2.1 Offset 20h: USB*2.0_CMD—USB 2.0 Command Register 574
10.2.2.2 Offset 24h: USB2.0_STS—USB 2.0 Status Register 577
10.2.2.3 Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register 580
10.2.2.4 Offset 2Ch: FRINDEX—Frame Index Register 582

10.2.2.5 Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment Register 583
10.2.2.6 Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register 584
10.2.2.7 Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register 585
10.2.2.8 Offset 60h: CONFIGFLAG—Configure Flag Register 586
10.2.2.9 Offset 64h: PORTSC—Port N Status and Control Register 587
10.2.3 USB* 2.0-Based Debug Port Registers 592
10.2.3.1 Offset A0h: CNTL_STS—Control/Status Register 593
10.2.3.2 Offset A4h: USBPID—USB PIDs Register 595
10.2.3.3 Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register 596
10.2.3.4 Offset B0h: CONFIG—Configuration Register 596

11.0 SMBus Controller Registers (B0:D31:F3) 597
11.1 PCI Configuration Registers (SMBus—B0:D31:F3) 597
11.1.1 SMBus Controller PCI Register Address Map 597
11.1.1.1 Offset 00h: Vendor Identification Register 598
11.1.1.2 Offset 02h: Device Identification Register 598
11.1.1.3 Offset 04h: PCI Command Register 599
11.1.1.4 Offset 06h: PCI Status Register 600
11.1.1.5 Offset 08h: Revision Identification Register 601
11.1.1.6 Offset 09h: Programming Interface Register 601
11.1.1.7 Offset 0Ah: Sub Class Code Register 601
11.1.1.8 Offset 0Bh: Base Class Code Register 602
11.1.1.9 Offset 10h: SMBus Memory Base Address 0 602
11.1.1.10 Offset 14h: SMBus Memory Base Address 1 603
11.1.1.11 Offset 20h: SMBus Base Address Register 603
11.1.1.12 Offset 2Ch: Subsystem Vendor Identification Register 604
11.1.1.13 Offset 2Eh: Subsystem Identification Register 604
11.1.1.14 Offset 3Ch: Interrupt Line Register 605
11.1.1.15 Offset 3Dh: Interrupt Pin Register 605
11.1.1.16 Offset 40h: Host Configuration Register 606
11.2 SMBus I/O and Memory Mapped I/O Registers 607
11.2.1 SMBus Registers 607
11.2.1.1 Offset 00h: Host Status Register 608
11.2.1.2 Offset 02h: Host Control Register 610
11.2.1.3 Offset 03h: Host Command Register 612
11.2.1.4 Offset 04h: Transmit Slave Address Register 612
11.2.1.5 Offset 05h: Host Data 0 Register 613
11.2.1.6 Offset 06h: Host Data 1 Register 613
11.2.1.7 Offset 07h: Host Block Data Byte Register 614
11.2.1.8 Offset 08h: Packet Error Check (PEC) Register 615
11.2.1.9 Offset 09h: Receive Slave Address Register 615
11.2.1.10 Offset 0Ah: Receive Slave Data Register 616
11.2.1.11 Offset 0Ch: Auxiliary Status Register 616
11.2.1.12 Offset 0Dh: Auxiliary Control Register 617
11.2.1.13 Offset 0Eh: SMLink Pin Control Register 618
11.2.1.14 Offset 0Fh: SMBus Pin Control Register 619
11.2.1.15 Offset 10h: Slave Status Register 620
11.2.1.16 Offset 11h: Slave Command Register 621
11.2.1.17 Offset 14h: Notify Device Address Register 622
11.2.1.18 Offset 16h: Notify Data Low Byte Register 622
11.2.1.19 Offset 17h: Notify Data High Byte Register 623

12.0 PCI Express* Configuration Registers (B0:D28:F0/1/2/3) 624
12.1 PCI Express* Configuration Registers (PCI Express* — B0:D28:F0/F1/F2/F3) 624
12.1.1 PCI Express* Configuration Registers Address Map 624
12.1.1.1 Offset 00h: Vendor Identification Register 627
12.1.1.2 Offset 02h: DID—Device Identification Register 627

12.1.1.3 Offset 04h: PCI COMMAND Register 628
12.1.1.4 Offset 06h: PCI Status Register 629
12.1.1.5 Offset 08h: RID—Revision Identification Register 631
12.1.1.6 Offset 09h: Programming Interface Register 631
12.1.1.7 Offset 0Ah: Sub Class Code Register 632
12.1.1.8 Offset 0Bh: Base Class Code Register 632
12.1.1.9 Offset 0Ch: Cache Line Size Register 633
12.1.1.10 Offset 0Dh: Primary Latency Timer Register 633
12.1.1.11 Offset 0Eh: Header Type Register 634
12.1.1.12 Offset 18h: Bus Number Register 634
12.1.1.13 Offset 1Bh: Secondary Latency Timer 635
12.1.1.14 Offset 1Ch: IOBL—I/O Base and Limit Register 635
12.1.1.15 Offset 1Eh: Secondary Status Register 636
12.1.1.16 Offset 20h: Memory Base and Limit Register 637
12.1.1.17 Offset 24h: Prefetchable Memory Base and Limit Register 638
12.1.1.18 Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register. 638
12.1.1.19 Offset 2Ch: Prefetchable Memory Limit Upper 32 Bits Register 639
12.1.1.20 Offset 34h: Capabilities List Pointer Register 639
12.1.1.21 Offset 3Ch: Interrupt Information Register 640
12.1.1.22 Offset 3Eh: Bridge Control Register 640
12.1.1.23 Offset 40h: Capabilities List Register 642
12.1.1.24 Offset 42h: PCI Express* Capabilities Register 642
12.1.1.25 Offset 44h: Device Capabilities Register 643
12.1.1.26 Offset 48h: Device Control Register 644
12.1.1.27 Offset 4Ah: Device Status Register 645
12.1.1.28 Offset 4Ch: Link Capabilities Register 646
12.1.1.29 Offset 50h: Link Control Register 648
12.1.1.30 Offset 52h: Link Status Register 649
12.1.1.31 Offset 54h: Slot Capabilities Register 650
12.1.1.32 Offset 58h: Slot Control Register 651
12.1.1.33 Offset 5Ah: Slot Status Register 652
12.1.1.34 Offset 5Ch: Root Control Register 653
12.1.1.35 Offset 60h: Root Status Register 654
12.1.1.36 Offset 64h: Device Capabilities 2 Register 654
12.1.1.37 Offset 68h: Device Control 2 Register 655
12.1.1.38 Offset 70h: Link Control 2 Register 656
12.1.1.39 Offset 80h: Message Signaled Interrupt Identifiers Register 656
12.1.1.40 Offset 82h: Message Signaled Interrupt Message Control Register 657
12.1.1.41 Offset 84h: Message Signaled Interrupt Message Address Register 658
12.1.1.42 Offset 88h: Message Signaled Interrupt Message Data Register 658
12.1.1.43 Offset 90h: Subsystem Vendor Capability Register 659
12.1.1.44 Offset 94h: Subsystem Vendor Identification Register 659
12.1.1.45 Offset A0h: Power Management Capability Register 660
12.1.1.46 Offset A2h: PCI Power Management Capabilities Register 660
12.1.1.47 Offset A4h: PCI Power Management Control and Status Register 661
12.1.1.48 Offset D4h: Miscellaneous Port Configuration Register 2 662
12.1.1.49 Offset D8h: Miscellaneous Port Configuration Register 663
12.1.1.50 Offset DCh: SMI/SCI Status Register 666
12.1.1.51 Offset E1h: Root Port Dynamic Clock Gating Enable 667
12.1.1.52 Offset E8h: PCI Express* Configuration Register 1 668
12.1.1.53 Offset 104h: Uncorrectable Error Status Register 668
12.1.1.54 Offset 108h: Uncorrectable Error Mask 670
12.1.1.55 Offset 10Ch: Uncorrectable Error Severity 672
12.1.1.56 Offset 110h: Correctable Error Status Register 673
12.1.1.57 Offset 114h: Correctable Error Mask Register 674
12.1.1.58 Offset 118h: Advanced Error Capabilities and Control Register 675
12.1.1.59 Offset 130h: Root Error Status Register 676
12.1.1.60 Offset 180h: Root Complex Topology Capability List Register 677
12.1.1.61 Offset 184h: Element Self Description Register 677
12.1.1.62 Offset 190h: Upstream Link Description Register 678

12.1.1.63 Offset 198h: Upstream Link Base Address Register 679
12.1.1.64 Offset 300h: PCI Express* Configuration Register 2 679
12.1.1.65 Offset 318h: PCI Express* Extended Test Mode Register 680
12.1.1.66 Offset 324h: PCI Express* Configuration Register 1 680

13.0 High Precision Event Timer Registers 681
13.1 High Precision Behavioral Rules 681
13.1.1 Memory Mapped Registers 681
13.1.1.1 Offset 00h: General Capabilities and Identification Register 682
13.1.1.2 Offset 010h: General Configuration Register 683
13.1.1.3 Offset 020h: General Interrupt Status Register 684
13.1.1.4 Offset 0F0h: Main Counter Value Register 685
13.1.1.5 Offset 100h: Timer n Configuration and Capabilities Register 686
13.1.1.6 Offset 108h: Timer n Comparator Value Register 691

14.0 Serial Peripheral Interface (SPI) 692
14.1 Serial Peripheral Interface Memory Mapped Configuration Registers 692
14.1.1 Serial Peripheral Interface (SPI) Register Address Map 692
14.1.1.1 Offset 00h: BIOS Flash Primary Region Register (SPI Memory Mapped Configuration Registers) 694
14.1.1.2 Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) 695
14.1.1.3 Offset 06h: Hardware Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers) 697
14.1.1.4 Offset 08h: Flash Address Register (SPI Memory Mapped Configuration Registers) 698
14.1.1.5 Offset 10h: Flash Data 0 Register (SPI Memory Mapped Configuration Registers) 699
14.1.1.6 Offset 14h: Flash Data [N] Register (SPI Memory Mapped Configuration Registers) 699
14.1.1.7 Offset 50h: Flash Regions Access Permissions Register (SPI Memory Mapped Configuration Registers) 700
14.1.1.8 Offset 54h: Flash Region 0 (Flash Descriptor) Register (SPI Memory Mapped Configuration Registers) 701
14.1.1.9 Offset 58h: Offset 00h: Flash Region 1 (BIOS Descriptor) Register (SPI Memory Mapped Configuration Registers) 701
14.1.1.10 Offset 5Ch: Flash Region 2 (Intel ME) Register (SPI Memory Mapped Configuration Registers) 702
14.1.1.11 Offset 60h: Flash Region 3 Register (SPI Memory Mapped Configuration Registers) 702
14.1.1.12 Offset 64h: Flash Region 4 (Platform Data) Register (SPI Memory Mapped Configuration Registers) 703
14.1.1.13 Offset 74h: Protected Range 0 Register (SPI Memory Mapped Configuration Registers) 703
14.1.1.14 Offset 78h: Protected Range 1 Register (SPI Memory Mapped Configuration Registers) 704
14.1.1.15 Offset 7Ch: Protected Range 2 Register (SPI Memory Mapped Configuration Registers) 705
14.1.1.16 Offset 80h: Protected Range 3 Register (SPI Memory Mapped Configuration Registers) 706
14.1.1.17 Offset 84h: Protected Range 4 Register (SPI Memory Mapped Configuration Registers) 707
14.1.1.18 Offset 90h: Software Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) 708
14.1.1.19 Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers) 709
14.1.1.20 Offset 94h: Prefix Opcode Configuration Register (SPI Memory Mapped Configuration Registers) 711
14.1.1.21 Offset 96h: Opcode Type Configuration Register (SPI Memory Mapped Configuration Registers) 712

14.1.1.22 Offset 98h: Opcode Menu Configuration Register (SPI Memory Mapped Configuration Registers) 713
14.1.1.23 Offset A0h: BIOS Base Address Configuration Register (SPI Memory Mapped Configuration Registers) 714
14.1.1.24 Offset B0h: Flash Descriptor Observability Control Register (SPI Memory Mapped Configuration Registers) 715
14.1.1.25 Offset B4h: Flash Descriptor Observability Data Register (SPI Memory Mapped Configuration Registers) 715
14.1.1.26 Offset C0h: Additional Flash Control Register (SPI Memory Mapped Configuration Registers) 716
14.1.1.27 Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) 717
14.1.1.28 Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) 719
14.1.1.29 Offset D0h: Flash Partition Boundary (SPI Memory Mapped Configuration Registers) 721
14.2 Flash Descriptor Records 721
14.2.1 OEM Section 721
15.0 UART / WDT (SIW) 722
15.1 Overview 722
15.2 LPC Logical Devices 4 and 5: Serial Ports (UART1 and UART2) 722
15.2.1 UART Register Details 722
15.2.1.1 Offset 00h: RBR—Receive Buffer Register 724
15.2.1.2 Offset 00h: THR—Transmit Holding Register 724
15.2.1.3 Offset 01h: IER—Interrupt Enable Register 725
15.2.1.4 Offset 02h: IIR—Interrupt Identification Register 726
15.2.1.5 Offset 02h: FCR—FIFO Control Register 728
15.2.1.6 Offset 03h: LCR—Line Control Register 729
15.2.1.7 Offset 04h: MCR—Modem Control Register 731
15.2.1.8 Offset 05h: LSR—Line Status Register 732
15.2.1.9 Offset 06h: MSR—Modem Status Register 736
15.2.1.10 Offset 07h: SCR—Scratchpad Register 737
15.2.1.11 Offset 00h: DLL—Programmable Baud Rate Generator Divisor Latch Register Low 737
15.2.1.12 Offset 01h: DLH—Programmable Baud Rate Generator Divisor Latch Register High 738
15.3 Logical Device 6: Watchdog Timer 738
15.3.1 Watchdog Timer Register Details 739
15.3.1.1 Offset 00h: PV1R0—Preload Value 1 Register 0 739
15.3.1.2 Offset 01h: PV1R1—Preload Value 1 Register 1 740
15.3.1.3 Offset 02h: PV1R2—Preload Value 1 Register 2 740
15.3.1.4 Offset 04h: PV2R0—Preload Value 2 Register 0 741
15.3.1.5 Offset 05h: PV2R1—Preload Value 2 Register 1 741
15.3.1.6 Offset 06h: PV2R2—Preload Value 2 Register 2 742
15.3.1.7 Offset 08h: GISR—General Interrupt Status Register 742
15.3.1.8 Offset 0Ch: RR0—Reload Register 0 743
15.3.1.9 Offset 0Dh: RR1—Reload Register 1 743
15.3.1.10 Offset 10h: WDTCR—WDT Configuration Register 744
15.3.1.11 Offset 18h: WDTLR—WDT Lock Register 745
15.4 SIW Configuration 746
15.4.1 SIW Configuration Register Summary 746
15.4.1.1 Global Control/Configuration Registers [00h - 2Fh] 747
15.4.1.2 Logical Device Configuration Registers [30h — FFh] 748
16.0 Per Thread WDT (B0:D31:F7) 753
16.1 PCI Registers 753
16.2 Register Attribute Legends 753
16.3 Configuration Space 754

16.3.1 PCI Header (B0:D31:F7) 754
16.3.1.1 Offset 00h: ID—Identifiers 754
16.3.1.2 Offset 04h: CMD—Command 755
16.3.1.3 Offset 06h: DSTS—Device Status 756
16.3.1.4 Offset 08h: RID—Revision ID 757
16.3.1.5 Offset 09h: CC—Class Code 757
16.3.1.6 Offset 0Ch: CLS—Cache Line Size 758
16.3.1.7 Offset 0Dh: MLT—Master Latency Timer 758
16.3.1.8 Offset 0Eh: HTYPE—Header Type 758
16.3.1.9 Offset 10h: TBAR—WDT Table Base Address 759
16.3.1.10 Offset 14h: PBAR—WDT PBA Base Address 759
16.3.1.11 Offset 18h: CBAR—WDT CFG Base Address 760
16.3.1.12 Offset 2Ch: SSVID—Subsystem Vendor ID 760
16.3.1.13 Offset 2Eh: SSID Subsystem ID 761
16.3.1.14 Offset 34h: CAP—Capabilities Pointer 761
16.3.2 PCI MSI-X Capability 761
16.3.2.1 Offset 80h: MID—MSI-X Capability ID 762
16.3.2.2 Offset 82h: MC—MSI-X Control & Status 762
16.3.2.3 Offset 84h: MT—MSI-X Table 762
16.3.2.4 Offset 88h: MP—MSI-X Message PBA 763
16.4 MMIO Space 763
16.4.1 Table BAR Range 763
16.4.1.1 Offset 00h: WDT0MAW—WDT 0 Msg Add WARN 765
16.4.1.2 Offset 04h: WDT0MUAW—WDT 0 Msg Upper Add WARN 765
16.4.1.3 Offset 08h: WDT0MDW—WDT 0 Msg Data WARN 766
16.4.1.4 Offset 0Ch: WDT0VCW—WDT 0 Vector Ctrl WARN 766
16.4.1.5 Offset 10h: WDT0MAR—WDT 0 Msg Add RESET 766
16.4.1.6 Offset 14h: WDT0MUAR—WDT 0 Msg Upper Add RESET 766
16.4.1.7 Offset 18h: WDT0MDR—WDT 0 Msg Data RESET 767
16.4.1.8 Offset 1Ch: WDT0VCR—WDT 0 Vector Ctrl RESET 767
16.4.2 PBA BAR Range 768
16.4.2.1 Offset 00h: WDT_PBA0—Pending Bit Array 0 769
16.4.2.2 Offset 04h: WDT_PBA1—Pending Bit Array 1 770
16.4.2.3 Offset 08h: WDT_PBA2—Pending Bit Array 2 771
16.4.2.4 Offset 0Ch: WDT_PBA3—Pending Bit Array 3 772
16.4.3 CFG BAR Range 773
16.4.3.1 Offset 00h: WDT0COUNT—WDT Count Register 773
16.4.3.2 Offset 04h: WDT0CMD—WDT Cmd/Status Register 774
16.4.3.3 Offset 200h: WDT_PSCALE 774
16.4.3.4 Offset 204h: WDT_GBLCFG 775

17.0 Thermal Sensor Registers (B0:D31:F6) 776
17.1 Thermal Configuration Registers 776
17.1.1 PCI Configuration Topology 776
17.1.2 Configuration Register Access Restrictions 776
17.1.3 Temperature Alert 776
17.1.4 Register Attribute Legend 777
17.2 PCI Configuration Registers (Thermal Sensor - B0:D31:F6) 777
17.2.1 Thermal Sensor PCI Register Address Map 777
17.2.1.1 Offset 00h: VID—Vendor Identification 779
17.2.1.2 Offset 02h: DID—Device Identification 779
17.2.1.3 Offset 04h: CMD—Command 780
17.2.1.4 Offset 06h: STS—Status 781
17.2.1.5 Offset 08h: RID—Revision Identification 782
17.2.1.6 Offset 09h: PI—Programming Interface 782
17.2.1.7 Offset 0Ah: SCC—Sub Class Code 782
17.2.1.8 Offset 0Bh: BCC—Base Class Code 783
17.2.1.9 Offset 0Ch: CLS—Cache Line Size 783
17.2.1.10 Offset 0Dh: LT—Latency Timer 783

17.2.1.11 Offset 0Eh: HTYPE—Header Type....784
17.2.1.12 Offset 10h: TBAR—Thermal Base 784
17.2.1.13 Offset 14h: TBARH—Thermal Base High DWord 785
17.2.1.14 Offset 2Ch: SVID—Subsystem Vendor ID 785
17.2.1.15 Offset 2Eh: SID—Subsystem ID 786
17.2.1.16 Offset 34h: CP—Capabilities Pointer....786
17.2.1.17 Offset 3Ch: INTLN—Interrupt Line 787
17.2.1.18 Offset 3Dh: INTPN—Interrupt Pin 787
17.2.1.19 Offset 40h: TBARB—BIOS Assigned Thermal Base Address 788
17.2.1.20 Offset 44h: TBARBH—BIOS Assigned Thermal Base High DWord 788
17.2.1.21 Offset 50h: PID—PCI Power Management Capability ID 789
17.2.1.22 Offset 52h: PC—Power Management Capabilities 789
17.2.1.23 Offset 54h: PCS—Power Management Control And Status 790
17.2.1.24 Offset 80h: MID—Message Signaled Interrupt Identifiers 791
17.2.1.25 Offset 82h: MC—Message Signaled Interrupt Message Control....791
17.2.1.26 Offset 84h: MA—Message Signaled Interrupt Message Address....792
17.2.1.27 Offset 88h: MD—Message Signaled Interrupt Message Data 792
17.2.1.28 Offset 94h: CCTS0RD—Thermal Sensor 0 Remote Diode....792
17.2.1.29 Offset 96h: CCTS0SSI—Thermal Sensor 0 Sense Stage Inputs 793
17.2.1.30 Offset 98h: CCTS0DSAC0—Thermal Sensor 0 Delta Sigma ADC Control 0....794
17.2.1.31 Offset 9Ch: CCTS0DSAC 1—Thermal Sensor 0 Delta Sigma ADC Control 1....794
17.2.1.32 Offset A0h: CCTS0C—Thermal Sensor 0 Calibration 795
17.2.1.33 Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0 795
17.2.1.34 Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1 797
17.2.1.35 Offset ACh: CCTS0BP—Thermal Sensor 0 Bandgap Pins....798
17.2.1.36 Offset B4h: CCTS1RD—Thermal Sensor 1 Remote Diode....798
17.2.1.37 Offset B6h: CCTS1SSI—Thermal Sensor 1 Sense Stage Inputs 799
17.2.1.38 Offset B8h: CCTS1DSAC0—Thermal Sensor 1 Delta Sigma ADC Control 0....799
17.2.1.39 Offset BCh: CCTS1DSAC1—Thermal Sensor 1 Delta Sigma ADC Control 1....800
17.2.1.40 Offset C0h: CCTS1C—Thermal Sensor 1 Calibration 800
17.2.1.41 Offset C4h: CCTS1TL0—Thermal Sensor 1 Top Logic 0 801
17.2.1.42 Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1 801
17.2.1.43 Offset CCh: CCTS1BP—Thermal Sensor 1 Bandgap Pins....803
17.2.1.44 Offset F0h: Reserved 803
17.2.1.45 Offset F8h: MANID—Manufacturing/Process 803
17.2.2 PCH Thermal Reporting 804
17.2.2.1 DIMM Thermal Reporting Configurations....804
17.2.3 Thermal Reporting Registers 804
17.2.3.1 Offset 00h: TS0IU—Thermal Sensor 0 In Use....806
17.2.3.2 Offset 01h: TS0C—Thermal Sensor 0 Control 807
17.2.3.3 Offset 02h: TS0S—Thermal Sensor 0 Status 809
17.2.3.4 Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read....810
17.2.3.5 Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point 810
17.2.3.6 Offset 08h: TS0CO—Thermal Sensor 0 Calibration Offset....812
17.2.3.7 Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd 812
17.2.3.8 Offset 0Ch: TS0ES—Thermal Sensor 0 Error Status 813
17.2.3.9 Offset 0Dh: TS0GPEN—Thermal Sensor 0 General Purpose Event Enables....814
17.2.3.10 Offset 0Eh: TS0PC—Thermal Sensor 0 Policy Control....815
17.2.3.11 Offset 10h: CPEC—CPU Power Error Correction Data 816
17.2.3.12 Offset 12h: C0TA—CPU0 Temperature Adjust....816
17.2.3.13 Offset 16h: C1TA—CPU1 Temperature Adjust....817
17.2.3.14 Offset 1Ah: MC—MPC Control 817
17.2.3.15 Offset 22h: Timestamp 818
17.2.3.16 Offset 24h: DIMMEN—DIMM Enable 819

17.2.3.17 Offset 30h: C0TV—CPU0 Temperature Value 820
17.2.3.18 Offset 32h: C1TV—CPU1 Temperature Value 821
17.2.3.19 Offset 34h: C0EV—CPU0 Energy Value..... 821
17.2.3.20 Offset 38h: C1EV—CPU1 Energy Value..... 821
17.2.3.21 Offset 3Fh: AE—Alert Enable 822
17.2.3.22 Offset 40h: TS1IU—Thermal Sensor 1 In Use 823
17.2.3.23 Offset 41h: TS1C—Thermal Sensor 1 Control..... 823
17.2.3.24 Offset 42h: TS1S—Thermal Sensor 1 Status..... 825
17.2.3.25 Offset 43h: TS1TR—Thermal Sensor 1 Thermometer Read 826
17.2.3.26 Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point..... 826
17.2.3.27 Offset 48h: TS1CO—Thermal Sensor 1 Calibration Offset 828
17.2.3.28 Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd 829
17.2.3.29 Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status 829
17.2.3.30 Offset 4Dh: TS1GPEN—Thermal Sensor 1 General Purpose Event Enables 831
17.2.3.31 Offset 4Eh: TS1PC—Thermal Sensor 1 Policy Control 832
17.2.3.32 Offset 50h: HTS—HOST Turbo Status 833
17.2.3.33 Offset 56h: MTL—MCP Temperature Limit 833
17.2.3.34 Offset 58h: MTV—MCH Temperature Value..... 834
17.2.3.35 Offset 60h: MCPTV—MCP Temperature Value..... 834
17.2.3.36 Offset 64h: MMPC—Max MCH Power Clamp 834
17.2.3.37 Offset 66h: MMCPPC—Max MCP Power Clamp 835
17.2.3.38 Offset 82h: TS0PIEN—Thermal Sensor 0 PCI Interrupt Event Enables 835
17.2.3.39 Offset 83h: TS0LOCK—Thermal Sensor 0 Register Lock Controls 836
17.2.3.40 Offset 98h: STS—SMBus Turbo Status..... 836
17.2.3.41 Offset 9Ch: SEC—SMBus Event Clear..... 837
17.2.3.42 Offset A4h: TC3—Thermal Compares 3 837
17.2.3.43 Offset A8h: TC1—Thermal Compares 1 838
17.2.3.44 Offset ACh: TC2—Thermal Compares 2..... 838
17.2.3.45 Offset B0h: DIMM0—CPU0 DIMM Data..... 839
17.2.3.46 Offset B4h: DIMM1—CPU1 DIMM Data..... 841
17.2.3.47 Offset B8h: DIMMID—DIMM ID..... 842
17.2.3.48 Offset BCh: ECPCLAMP—EC Power Clamp Data 844
17.2.3.49 Offset C2h: TS1PIEN—Thermal Sensor 1 PCI Interrupt Event Enables 844
17.2.3.50 Offset C3h: TS1LOCK—Thermal Sensor 1 Register Lock Controls..... 845
17.2.3.51 Offset D8h: ITV—Internal Temperature Values..... 845

18.0 Intel® Management Engine Interface (MEI - B0:D22:F0/F1) 846
18.1 First Intel Management Engine Interface (MEI1) Configuration Registers (MEI1 — B0:D22:F0) 846
18.1.1 VID—Vendor Identification Register 847
18.1.2 DID—Device Identification Register 847
18.1.3 PCICMD—PCI Command Register 848
18.1.4 PCISTS—PCI Status Register 848
18.1.5 RID—Revision Identification Register 849
18.1.6 CC—Class Code Register 849
18.1.7 HTYPE—Header Type Register..... 849
18.1.8 MEI1_MBAR—MEI1 MMIO Base Address Register 850
18.1.9 SVID—Subsystem Vendor ID Register..... 850
18.1.10 SID—Subsystem ID Register 850
18.1.11 CAPP—Capabilities List Pointer Register..... 850
18.1.12 INTR—Interrupt Information Register..... 851
18.1.13 MLMG—Maximum Latency/Minimum Grant Register 851
18.1.14 HFS—Host Firmware Status Register..... 851
18.1.15 ME_UMA—Management Engine UMA Register..... 852
18.1.16 GMES—General ME Status..... 852
18.1.17 H_GS—Host General Status..... 852

18.1.18 PID—PCI Power Management Capability ID Register 853
18.1.19 PC—PCI Power Management Capabilities Register 853
18.1.20 PMCS—PCI Power Management Control and Status Register 854
18.1.21 MID—Message Signaled Interrupt Identifiers Register 854
18.1.22 MC—Message Signaled Interrupt Message Control Register 855
18.1.23 MA—Message Signaled Interrupt Message Address Register 855
18.1.24 MUA—Message Signaled Interrupt Upper Address Register 855
18.1.25 MD—Message Signaled Interrupt Message Data Register 855
18.1.26 HIDM—MEI Interrupt Delivery Mode 856
18.1.27 HERES—MEI Extend Register Status 856
18.1.28 HERX—MEI Extend Register DWX 857
18.2 MEI1_MBAR: MEI1 MMIO Registers 857
18.2.1 H_CB_WW—Host Circular Buffer Write Window 857
18.2.2 H_CSR—Host Control Status 858
18.2.3 ME_CB_RW—ME Circular Buffer Read Window 858
18.2.4 ME CSR_HA—ME Control Status Host Access 859
18.3 Second Management Engine Interface (MEI2) Configuration Registers (MEI2—B0:D22:F1) 860
18.3.1 VID—Vendor Identification Register 860
18.3.2 DID—Device Identification Register 861
18.3.3 PCICMD—PCI Command Register 861
18.3.4 PCISTS—PCI Status Register 862
18.3.5 RID—Revision Identification Register 862
18.3.6 CC—Class Code Register 862
18.3.7 HTYPE—Header Type Register 862
18.3.8 MEI_MBAR—MEI MMIO Base Address Register 863
18.3.9 SVID—Subsystem Vendor ID Register 863
18.3.10 SID—Subsystem ID Register 863
18.3.11 CAPP—Capabilities List Pointer Register 863
18.3.12 INTR—Interrupt Information Register 864
18.3.13 MLMG—Maximum Latency/Minimum Grant Register 864
18.3.14 HFS—Host Firmware Status Register 864
18.3.15 GMES—General ME Status 864
18.3.16 H_GS—Host General Status 865
18.3.17 PID—PCI Power Management Capability ID Register 865
18.3.18 PC—PCI Power Management Capabilities Register 865
18.3.19 PMCS—PCI Power Management Control and Status Register 866
18.3.20 MID—Message Signaled Interrupt Identifiers Register 866
18.3.21 MC—Message Signaled Interrupt Message Control Register 867
18.3.22 MA—Message Signaled Interrupt Message Address Register 867
18.3.23 MUA—Message Signaled Interrupt Upper Address Register 867
18.3.24 MD—Message Signaled Interrupt Message Data Register 867
18.3.25 HIDM—MEI Interrupt Delivery Mode 868
18.3.26 HERES—MEI Extend Register Status 868
18.3.27 HERX—MEI Extend Register DWX 869
18.4 MEI2_MBAR—MEI2 MMIO Registers 869
18.4.1 H_CB_WW—Host Circular Buffer Write Window 869
18.4.2 H_CSR—Host Control Status 870
18.4.3 ME_CB_RW—ME Circular Buffer Read Window 870
18.4.4 ME CSR_HA—ME Control Status Host Access 871

PCIe* EndPoint & GbE-Volume 2 of 4 872

19.0 PCI Express* EndPoint Introduction 874
19.1 PCI Express* EndPoint (EP) 874
19.1.1 Overview 874

19.1.2 Feature List 874
19.1.3 EP Interface Block Diagram 875
19.1.4 EP Functional Description 876
19.1.4.1 Transmit Interface (TI) 876
19.1.4.2 Receive Interface (RI) 876
19.1.4.3 PCIe* EP Functions 877
19.1.4.4 EP Function Mapping 877
19.1.4.5 EP Mapping of BARs to MMIO 878
19.1.5 PCIe* EP Interrupts 878
19.1.5.1 INTx 878
19.1.5.2 MSI-X 878
19.1.6 PCIe* EP Errors 879
19.1.6.1 PCIe* Error Management 879
19.1.6.2 PCIe* Error Reporting Mechanisms 879
19.1.6.3 PCIe* Error Handling and Signalling 880
19.1.6.4 PCIe* Error Sources 880
19.1.7 Device Specific Error Management 884
19.1.7.1 Memory Error Poisoning 884
19.1.8 PCIe* Request Generation 884
19.1.8.1 64-bit Memory Accesses 886
19.1.8.2 Completions 886
19.1.8.3 Messages 888
19.2 Intel® QuickAssist Technology 889
19.2.1 Features 889
19.2.2 EP Intel® QuickAssist Integrated Accelerator (IQIA) 890
19.2.3 Ring Controller 890
19.2.3.1 Features 890
19.2.3.2 Functionality 890

20.0 PCIe Endpoint Function 0 Registers 892
20.1 EP PF PCI Configuration Space 892
20.2 Detailed Register Summaries 894
20.2.1 PCI Views 894
20.2.2 PCI Standard Header Registers 899
20.2.2.1 PVID—PF Vendor Identification Register 899
20.2.2.2 PDID—PF Device Identification Register 899
20.2.2.3 PPCICMD—PF Device Command Register 900
20.2.2.4 PPCISTS—PF Device Status Register 901
20.2.2.5 PCC—PF Class Code Register 903
20.2.2.6 PHDR—PF Header Type Register 904
20.2.2.7 PeQATLBAR—PF QAT Lower Base Address Register 904
20.2.2.8 QATUBAR—PF QAT Upper Base Address Register 905
20.2.2.9 PMISCLBAR—PF Miscellaneous Lower Base Address Register 905
20.2.2.10 PMISCUBAR—PF Miscellaneous Upper Base Address Register 906
20.2.2.11 PETRINGCSRLBAR—PF Ring CSR Lower Base Address Register 906
20.2.2.12 PETRINGCSRUBAR—PF Ring CSR Upper Base Address Register 906
20.2.2.13 PSVID—PF Subsystem Vendor ID Register 907
20.2.2.14 PSID—PF Subsystem ID Register 907
20.2.2.15 PCP—PF Capabilities Pointer Register 908
20.2.2.16 PIRQL—PF Interrupt Line Register 908
20.2.2.17 PIRQP—PF Interrupt Pin Register 909
20.2.2.18 SKU: SKU Register 909
20.2.2.19 SKU Register 2 910
20.2.3 MSI-X Capability Structure 911
20.2.3.1 PMSI-X—PF Message Signalled Interrupt X Capability ID Register 911
20.2.3.2 PMSIXNCP—PF MSIX Next Capability Pointer Register 911
20.2.3.3 PMSIXCNTL—PF Message Signalled Interrupt X Control Register 912
20.2.3.4 PMSIXTBIR—PF MSI-X Table Offset & Table BIR Register 912

20.2.3.5 PMSIXPBABIR—PF MSI-X Pending Bit Array & BIR Offset Register 913
20.2.4 Power Management Capability Structure 913
20.2.4.1 PPMCAP—PF Power Management Capabilities ID Register 913
20.2.4.2 PPMCP—PF Power Management Next Capability Pointer Register 914
20.2.4.3 PPMC—PF Power Management Capabilities Register 914
20.2.4.4 PPMCSR—PF Power Management Control and Status Register 915
20.2.5 PCI Express Capability Structure 917
20.2.5.1 PPCID—PF PCI Express Capability ID Register 917
20.2.5.2 PPCP—PF PCI Express Next Capability Pointer Register 917
20.2.5.3 PPCR—PF PCI Express Capabilities Register 918
20.2.5.4 PPDCAP—PF PCI Express Device Capabilities Register 918
20.2.5.5 PPDCNTL—PF PCI Express Device Control Register 920
20.2.5.6 PPDSTAT—PF PCI Express Device Status Register 921
20.2.5.7 PLCAPR—PF Link Capabilities Register 922
20.2.5.8 PLCNTLR—PF Link Control Register 924
20.2.5.9 PLSR—PF Link Status Register 926
20.2.5.10 PDCAPR2—PF Device Capabilities 2 Register 927
20.2.5.11 PDCNTR2—PF Device Control 2 Register 928
20.2.5.12 PLCNTLR2—PF Link Control 2 Register 929
20.2.5.13 PLSR2—PF Link Status 2 Register 931
20.2.6 MSI Capability Structure 931
20.2.6.1 PMSICID—PF Message Signalled Interrupt Capability ID Register 931
20.2.6.2 PMSINCP—PF Message Signalled Interrupt Next Capability Pointer Register 932
20.2.6.3 PMSICTL—PF Message Signalled Interrupt Control Register 932
20.2.6.4 PMSILADDR—PF Message Signalled Interrupt Lower Address Register 933
20.2.6.5 PMSIUADDR—PF Message Signalled Interrupt Upper Address Register 933
20.2.6.6 PMSIDATA—PF Message Signalled Interrupt Data Register 934
20.2.6.7 PMSIMSK—PF Message Signalled Interrupt Mask Register 934
20.2.6.8 PMSIPND—PF Message Signalled Interrupt Pending Register 935
20.2.7 PF Advanced Error Reporting Capability Structure 935
20.2.7.1 PPCIEAERCAPID—PF PCI Express AER Capability ID Register 935
20.2.7.2 PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register 936
20.2.7.3 PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register 937
20.2.7.4 PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register 938
20.2.7.5 PPAERCS—PF PCI Express AER Correctable Error Register 939
20.2.7.6 PPAERCM—PF PCI Express AER Correctable Error Mask Register 940
20.2.7.7 PPAERCTLCAP—PF PCI Express AER Control and Capability Register 941
20.2.7.8 PPAERHDRLOG0—PF PCI Express AER Header Log 0 Register 941
20.2.7.9 PPAERHDRLOG1—PF PCI Express AER Header Log 1 Register 942
20.2.7.10 PPAERHDRLOG2—PF PCI Express AER Header Log 2 Register 942
20.2.7.11 PPAERHDRLOG3—PF PCI Express AER Header Log 3 Register 943
20.2.8 PF Alternative Routing-ID Extended Capability Structure 943
20.2.8.1 PARIDHDR—PF Alternative Routing ID Capability Header 943
20.2.8.2 PFARICAP—PF ARI Capabilities Register 944
20.2.8.3 PARIDCTL—PF Alternative Routing ID Control Register 945
20.2.9 PF SR-IOV Extended Capability Structure 945
20.2.9.1 PSRIOVCAPID—PF SR-IOV Capability ID Register 945
20.2.9.2 PSRIOVCVNC—PF SRIOV Capability Version and Next Capability Pointer Register 946
20.2.9.3 PSRIOVCAP—PF SRIOV Capabilities Register 947
20.2.9.4 PSRIOVCS—PF SRIOV Control and Status Register 947
20.2.9.5 PSRIOVMTOTINI—PF SRIOV Initial and Total VFs Register 947
20.2.9.6 PSRIOVNUMVF—PF SRIOV Number of VFs Register 948
20.2.9.7 PSRIOVFVFO—PF SRIOV First VF Offset Register 948
20.2.9.8 PSRIOVVFS—PF SRIOV VF Stride Register 949
20.2.9.9 PSRIOVFDID—PF SRIOV VF Device ID Register 949
20.2.9.10 PSRIOVPAGESIZE—PF SRIOV Supported Page Size Register 950

20.2.9.11 PSRIOVSYSPS—PF SRIOV System Page Size Register 950
20.2.9.12 PSRIOVLBAR0—SRIOV Lower BAR0 Register..... 951
20.2.9.13 PSRIOVUBAR0—SRIOV Upper BAR0 Register 952
20.2.9.14 PSRIOVLBAR1—SRIOV Lower BAR1 Register..... 953
20.2.9.15 PSRIOVUBAR1—SRIOV Upper BAR1 Register 954
20.2.9.16 PSRIOVVFMA—PF SRIOV VF Migration Array Register..... 955
20.3 EP VF PCI Configuration Space 956
20.3.1 PCI Standard Header Registers 956
20.3.2 VF PCI Express Capability Structure..... 962
20.3.2.10 VDCAPR2[0:15]—VF Device Capabilities 2 Register..... 969
20.3.3 VF MSI Capability Structure 970
20.3.3.7 VMSIMSK—VF Message Signalled Interrupt Mask Register 972
20.3.3.8 VMSIPND—VF Message Signalled Interrupt Pending Register..... 973
20.3.4 VF Advanced Error Reporting Capability Structure..... 973
20.3.5 VF Alternative Routing ID Extended Capability Structure..... 983
20.4 EP Memory Mapped Registers 984
20.4.1 Detailed Register Summary 984
20.4.2 CSRs 984
20.4.2.1 ERRSOU2—Error Source Register 2 985
20.4.2.2 ERRMSK2—Error Source Mask Register 2 985
20.4.2.3 SINTPF—Signal Raw PF Interrupt Register..... 986
20.4.2.4 SMIAPF—Signal IA PF Interrupt Mask Register 986
20.4.2.5 GBECFGMMIOV—GBE Configuration and MMIO Valid Register..... 987
20.4.2.6 SINTGBE[0:3]—Signal Raw PF Interrupt GBE Register 988
20.4.2.7 SMIAGBE[0:3]—Signal IA Interrupt Mask GBE Register..... 988

21.0 GbE Controller Overview..... 990
21.1 Feature Summary 990
21.2 Scope 991
21.3 Terminology and Acronyms..... 991
21.3.1 External Specification and Documents..... 992
21.3.1.1 Network Interface Documents 993
21.3.1.2 Host Interface Documents..... 993
21.3.1.3 Networking Protocol Documents..... 993
21.3.1.4 Manageability Document..... 993
21.4 Product Overview 993
21.5 External Interface 994
21.5.1 PCIe* Interface (Connected to PCI Express Through PCH PCIe* End Point) 994
21.5.2 Network Interfaces..... 994
21.5.3 EEPROM Interface 995
21.5.4 SMBus Interface 995
21.5.5 MDIO/I^2C Two-Wire Interfaces..... 995
21.5.6 Software-Definable Pins (SDP) Interface (General-Purpose I/O)..... 995
21.5.7 LED Interface 996
21.6 GbE Controller Features and Capabilities 996
21.6.1 Network Interface 999
21.6.1.1 Quad Port 999
21.6.1.2 Shared MDIO Support 999
21.6.1.3 I^2C Clock Stretching..... 999
21.6.1.4 1000BASE-KX Backplane Ethernet Interface 999
21.6.2 HOST Interface 999
21.6.2.1 PCIe* Connectivity Through EndPoint Interface..... 999
21.6.2.2 64-bit BAR Support 999
21.6.3 Performance Features.....1000
21.6.4 Receive and Transmit Queues 1000
21.6.4.1 Unused Receive and Transmit Ports Buffer Sharing.....1000
21.6.5 Virtualization 1000

21.6.5.1 Malicious Driver Detection 1000
21.6.5.2 2-tuple Filtering 1000
21.6.6 Security Offload 1000
21.6.6.1 Transmit Queue Prioritization 1001
21.6.7 Manageability 1001
21.6.7.1 SMBus Interface to External BMC 1001
21.6.7.2 Exclusive Management Filtering 1001
21.6.8 Time SYNC (IEEE1588 and IEEE 802.1AS) 1001
21.6.8.1 Per Packet Timestamp 1001
21.6.8.2 Improved System Time Accuracy 1001
21.6.8.3 SYSTIM Synchronized Pulse Generation on SDP Pins 1001
21.6.8.4 Time SYNC Interrupts 1002
21.7 Device Data Flows 1002
21.7.1 Transmit Data Flow 1002
21.7.2 Rx Data Flow 1003

22.0 GbE Interconnects 1004
22.1 PCIe* Interface 1004
22.1.1 Special GbE Controller Features 1004
22.1.2 Relaxed Ordering 1004
22.1.3 Snoop Not Required 1005
22.1.4 No Snoop and Relaxed Ordering for LAN Traffic 1005
22.1.4.1 No-Snoop Option for Payload 1006
22.1.5 PCIe* Power Management 1006
22.2 Management Interfaces 1006
22.2.1 SMBus 1006
22.2.1.1 Channel Behavior 1006
22.3 EEPROM 1016
22.3.1 EEPROM Interface 1016
22.3.1.1 General Overview 1016
22.3.1.2 EEPROM Device 1017
22.3.1.3 HW Initial Load Process 1017
22.3.1.4 Software Accesses 1020
22.3.1.5 Signature Field 1021
22.3.1.6 EEPROM Recovery 1021
22.3.2 Shared EEPROM 1021
22.3.2.1 EEPROM Deadlock Avoidance 1021
22.3.2.2 EEPROM Map Shared Words 1022
22.3.3 Vital Product Data (VPD) Support 1022
22.4 Configurable I/O Pins 1023
22.4.1 General-Purpose I/O (Software-Definable Pins) 1023
22.4.2 Software Watchdog 1024
22.4.2.1 Watchdog Rearm 1024
22.4.3 LEDs 1024
22.5 Network Interfaces 1025
22.5.1 Overview 1025
22.5.2 MAC Functionality 1026
22.5.2.1 Internal GMII/MII Interface 1026
22.5.2.2 MDIO/MDC PHY Management Interface 1026
22.5.3 SerDes, SGMII and 1000BASE-KX Support 1028
22.5.3.1 SerDes, SGMII and 1000BASE-KX PCS Block 1028
22.5.3.2 GbE Physical Coding Sub-Layer (PCS) 1028
22.5.4 Auto-Negotiation and Link Setup Features 1029
22.5.4.1 SerDes Link Configuration 1030
22.5.4.2 1000BASE-KX Link Configuration 1032
22.5.4.3 SGMII Link Configuration 1033
22.5.4.4 Loss of Signal/Link Status Indication 1033

22.5.5 Ethernet Flow Control (FC) 1034
22.5.5.1 MAC Control Frames and Receiving Flow Control Packets 1035
22.5.5.2 PAUSE and MAC Control Frames Forwarding 1036
22.5.5.3 Transmission of PAUSE Frames 1036
22.5.5.4 IPG Control and Pacing 1038
22.5.6 Loopback Support 1038
22.5.6.1 General 1038
22.5.6.2 MAC Loopback 1038
22.5.6.3 SerDes, SGMII and 1000BASE-KX Loopback 1039

23.0 GbE Initialization 1040
23.1 Power Up 1040
23.1.1 Power-Up Sequence 1040
23.1.2 Power-Up Timing Diagram 1041
23.2 Reset Operation 1042
23.2.1 Hardware-Based Reset Sources 1042
23.2.1.1 GBE_AUX_PWR_OK 1042
23.2.1.2 PCIE_EP_RST# 1042
23.2.1.3 Function Level Reset (FLR) 1042
23.3 Software Based Reset Sources 1043
23.3.1 Full Software Reset (DEV_RST and RST) 1043
23.3.2 BME (Bus Master Enable) 1044
23.3.2.1 Transaction Flow for BME When GIO Master Disable Algorithm is Used 1044
23.3.2.2 Transaction Flow for BME When GIO Master Disable Algorithm is Not Used 1044
23.3.3 Force TCO 1045
23.3.4 Firmware Reset 1045
23.3.5 EEPROM Reset 1045
23.3.6 External Phy Reset 1045
23.3.7 CSR Access Flow Following Power-on and Reset Events 1045
23.3.8 Reset Effects 1046
23.4 Software Initialization and Diagnostics 1049
23.4.1 Power Up State 1050
23.4.2 Initialization Sequence 1050
23.4.3 Interrupts During Initialization 1050
23.4.4 Global Reset and General Configuration 1051
23.4.5 Flow Control Setup 1051
23.4.6 Link Setup Mechanisms and Control/Status Bit Summary 1051
23.4.6.1 MAC/SERDES Link Setup (CTRL_EXT.LINK_MODE = 011b) 1051
23.4.6.2 MAC/SGMII Link Setup (CTRL_EXT.LINK_MODE = 010b) 1052
23.4.6.3 MAC/1000BASE-KX Link Setup (CTRL_EXT.LINK_MODE = 001b) 1053
23.4.7 Initialization of Statistics 1053
23.4.8 Receive Initialization 1054
23.4.8.1 Initialize the Receive Control Register 1054
23.4.8.2 Dynamic Enabling and Disabling of Receive Queues 1054
23.4.9 Transmit Initialization 1055
23.4.9.1 Dynamic Queue Enabling and Disabling 1055
23.4.10 Virtualization Initialization Flow 1056
23.4.10.1 VMDq Mode 1056
23.5 Access to Shared Resources 1057
23.5.1 Acquiring Ownership Over a Shared Resource 1057
23.5.2 Releasing Ownership Over a Shared Resource 1058

24.0 Non-Volatile Memory Map - EEPROM 1059
24.1 EEPROM General Map 1059
24.2 EEPROM Configuration Notes 1062
24.3 Hardware Accessed Words 1063

24.3.1 Ethernet Address (LAN Base Address + Offsets 0x00-0x02) 1063
24.3.2 Initialization Control Word 1 (Word 0x0A) 1063
24.3.3 Subsystem ID (Word 0x0B) 1064
24.3.4 Subsystem Vendor ID (Word 0x0C) 1064
24.3.5 Device ID (LAN Base Address + Offset 0x0D) 1065
24.3.6 Vendor ID (Word 0x0E) 1065
24.3.7 Initialization Control Word 2 (Word 0x0F) 1065
24.3.8 EEPROM Sizing (Word 0x12) 1066
24.3.9 Initialization Control 4 (LAN Base Address + Offset 0x13) 1067
24.3.10 PHY_RST_CONTROL (LAN Base Address + Offset 0x16) 1068
24.3.11 Device Rev ID (Word 0x1E) 1068
24.3.12 LED 0 Configuration Defaults (LAN Base Address + Offset 0x1F) 1068
24.3.13 Software Defined Pins Control (LAN Base Address + Offset 0x20) 1069
24.3.14 CGB Functions Control (Word 0x21) 1070
24.3.15 Initialization Control 3 (LAN Base Address + Offset 0x24) 1071
24.3.16 PCIe* Control 3 (Word 0x29) 1072
24.3.17 Watchdog Configuration (Word 0x2E) 1073
24.3.18 VPD Pointer (Word 0x2F) 1073
24.4 CSR Auto Configuration Pointer (LAN Base Address + Offset 0x17) 1073
24.4.1 CSR Configuration Section Length - Offset 0x0 1074
24.4.2 Block CRC8 (Offset 0x1) 1074
24.4.3 CSR Address - Offset 0x2 1074
24.4.4 CSR Data LSB - Offset 0x3 1074
24.4.5 CSR Data MSB - Offset 0x4 1074
24.5 CSR Auto Configuration Power-Up Pointer (LAN Base Address + Offset 0x27) 1075
24.5.1 CSR Configuration Power-Up Section Length - Offset 0x0 1075
24.5.2 Block CRC8 (Offset 0x1) 1075
24.5.3 CSR Address - Offset 0x2 1076
24.5.4 CSR Data LSB - Offset 0x3 1076
24.5.5 CSR Data MSB - Offset 0x4 1076
24.5.6 EICT Information Structure 1076
24.6 PHY Configuration Structure 1077
24.6.1 PHY Configuration Section Length - Offset 0x0 1077
24.6.2 Block CRC8 (Offset 0x1) 1077
24.6.3 PHY Number and PHY Address - (Offset 2*n; [n = 1... Section Length]) 1078
24.6.4 PHY Data (Offset 2*n + 1; [n = 1... Section Length]) 1078
24.7 Pointers and Control Words Used By Firmware 1078
24.7.1 Pass Through LAN Configuration Pointer (LAN Base Address + Offset 0x11) .. 1078
24.7.2 PHY Configuration Pointer (Word 0x50) 1078
24.7.3 Firmware Patch Pointer (Word 0x51) 1079
24.7.4 Sideband Configuration Pointer (Word 0x57) 1079
24.7.5 Manageability Capability/Manageability Enable (Word 0x54) 1079
24.7.6 Management HW Config Control (Word 0x23) 1080
24.8 Firmware Patch Structure 1080
24.8.1 Firmware Patch Data Size (Offset 0x0) 1080
24.8.2 Block CRC8 (Offset 0x1) 1080
24.8.3 Patch Ram Address Word (Offset 0x2) 1081
24.8.4 Patch Version 1 Word (Offset 0x3) 1081
24.8.5 Patch Version 2 Word (Offset 0x4) 1081
24.8.6 Patch Version 3 Word (Offset 0x5) 1081
24.8.7 Patch Version 4 Word (Offset 0x6) 1081
24.8.8 Patch Data Words (Offset 0x7, Block Length) 1081
24.9 PT LAN Configuration Structure 1082
24.9.1 PT LAN Configuration Structure Section Length - Offset 0x0 1082
24.9.2 Block CRC8 (Offset 0x1) 1082

24.9.3 CSR Address - (Offset 2*n; [n = 1... Section Length]) 1082
24.9.4 CSR Data LSB - (Offset 0x1 + 2*n; [n = 1... Section Length]) 1083
24.9.5 CSR Data MSB - (Offset 0x2 + 2*n; [n = 1... Section Length])....1083
24.9.6 Manageability Filters....1083
24.10 Sideband Configuration Structure 1084
24.10.1 Section Length (Offset 0x0)....1084
24.10.2 Block CRC8 (Offset 0x1)....1084
24.10.3 SMBus Max Fragment Size (Offset 0x2)....1084
24.10.4 SMBus Notification Timeout (Offset 0x3) 1084
24.10.5 SMBus Slave Address 0 1 (Offset 0x4) 1084
24.10.6 SMBus Slave Address 2 3 (Offset 0x5) 1085
24.10.7 Reserved (Offset 0x6)....1085
24.10.8 LAN Receive Enable 1 (Offset 0x7) 1085
24.10.9 LAN Receive Enable 2 (Offset 0x8) 1085
24.10.10 SMBus Flags (Offset 0x9) 1086
24.10.11 LAN Receive Enable 3 (Offset 0xA) 1086
24.11 Software Accessed Words....1090
24.11.1 Compatibility (Word 0x03) 1090
24.11.2 Port Identification LED Blinking (Word 0x04)....1091
24.11.3 OEM Specific (Word 0x06, 0x07) 1091
24.11.4 EEPROM Image Revision (Word 0x05)....1091
24.11.5 PBA Number (Word 0x09) 1092
24.11.5.1 PBA Structure 1092
24.11.6 PXE Main Setup Options (Word 0x30/0x34/0x38/0x3A)....1092
24.11.7 PXE Configuration Customization Options (Word 0x31/0x35/0x39/0x3B) 1093
24.11.8 PXE Boot Agent Version Number (Word 0x32)....1094
24.11.9 IBA Capabilities (Word 0x33)....1094
24.11.10 PXE Reserved Words (Words 0x36, 0x3C, 0x3E)....1095
24.11.11 Alternate MAC Address Pointer (Word 0x37) 1095
24.11.12 Checksum Word (Offset 0x3F) 1095
24.11.13 Image Unique ID (Word 0x42, 0x43) 1095

25.0 GbE Power Management....1096
25.1 General Power State Information....1096
25.2 Power States 1096
25.2.1 D0 Uninitialized (D0u) State 1097
25.2.1.1 Entry Into D0u State 1097
25.2.2 D0 Active (D0a) State....1097
25.2.2.1 Entry to D0a State 1097
25.2.3 D3 State (PCI-PM D3hot) 1097
25.2.3.1 Entry to D3 State....1098
25.2.3.2 Master Disable Via CTRL Register 1098
25.2.4 Dr State (D3cold)....1099
25.2.4.1 Dr Disable Mode 1099
25.2.4.2 Entry to Dr State 1100
25.2.4.3 Auxiliary Power Usage 1100
25.2.5 Device Power-Down State 1100
25.3 Timing of Power-State Transitions 1100
25.3.1 Power Up (Off to Dup to D0u to D0a)....1101
25.3.2 Transition From D0a to D3 and Back Without PCIE_EP_RST# 1102
25.3.3 Transition From D0a to D3 and Back With PCIE_EP_RST# 1103
25.3.4 Transition From D0a to Dr and Back Without Transition to D3 1104
25.4 Wake Up 1105
25.4.1 Advanced Power Management Wake Up 1105
25.4.2 PCIe* Power Management Wake Up....1106

25.4.3 Wake-Up Packets 1107
25.4.3.1 Pre-Defined Filters 1107
25.4.3.2 Flexible Filters 1110
25.4.3.3 Wake Up Packet Storage 1112

26.0 GbE Inline Functions 1113
26.1 Receive Functionality 1113
26.1.1 Receive Queues Assignment 1113
26.1.1.1 Queuing in a Non-Virtualized Environment 1115
26.1.1.2 Receive Queuing in a Virtualized Environment 1115
26.1.1.3 Queue Configuration Registers 1116
26.1.1.4 L2 Ether-Type Filters 1117
26.1.1.5 Filters 1117
26.1.1.6 Flex Filters 1118
26.1.1.7 SYN Packet Filters 1119
26.1.1.8 Receive-Side Scaling (RSS) 1119
26.1.2 L2 Packet Filtering 1125
26.1.2.1 MAC Address Filtering 1126
26.1.2.2 VLAN Filtering 1128
26.1.2.3 Platform Manageability Filtering 1129
26.1.3 Receive Data Storage 1130
26.1.3.1 Host Buffers 1130
26.1.3.2 On-Chip Receive Buffers 1131
26.1.3.3 On-Chip Descriptor Buffers 1131
26.1.4 Legacy Receive Descriptor Format 1131
26.1.5 Advanced Receive Descriptors 1134
26.1.5.1 Advanced Receive Descriptors (RDESC) - Read Format 1134
26.1.5.2 Advanced Receive Descriptors (RDESC) - Writeback Format 1135
26.1.6 Receive Descriptor Fetching 1139
26.1.7 Receive Descriptor Write-Back 1139
26.1.8 Receive Descriptor Ring Structure 1140
26.1.8.1 Low Receive Descriptors Threshold 1142
26.1.9 Header Splitting and Replication 1142
26.1.9.1 Purpose 1142
26.1.9.2 Description 1143
26.1.10 Receive Packet Timestamp in Buffer 1145
26.1.11 Receive Packet Checksum Off Loading 1146
26.1.11.1 Filters Details 1147
26.1.11.2 Receive UDP Fragmentation Checksum 1149
26.1.12 SCTP Offload 1150
26.2 Transmit Functionality 1150
26.2.1 Packet Transmission 1150
26.2.1.1 Transmit Data Storage 1151
26.2.1.2 On-Chip Transmit Buffers 1151
26.2.1.3 On-Chip Descriptor Buffers 1151
26.2.1.4 Transmit Contexts 1151
26.2.2 Transmit Descriptors 1152
26.2.2.1 Legacy Transmit Descriptor Format 1153
26.2.2.2 Advanced Transmit Context Descriptor 1156
26.2.2.3 Advanced Transmit Data Descriptor 1158
26.2.2.4 Transmit Descriptor Ring Structure 1161
26.2.2.5 Transmit Descriptor Fetching 1163
26.2.2.6 Transmit Descriptor Write-Back 1163
26.2.3 Transmit Completions Head Write Back 1164
26.2.3.1 Description 1164
26.2.4 TCP/UDP Segmentation 1165
26.2.4.1 Assumptions 1166

26.2.4.2 Transmission Process 1166
26.2.4.3 TCP Segmentation Performance 1168
26.2.4.4 Packet Format 1168
26.2.4.5 TCP/UDP Segmentation Indication 1169
26.2.4.6 Transmit Checksum Offloading with TCP/UD Segmentation 1170
26.2.4.7 TCP/UDP/IP Header Update 1171
26.2.4.8 IP/TCP/UDP Checksum Offloading 1173
26.2.4.9 Data Flow 1173
26.2.5 Checksum Offloading in Non-Segmentation Mode 1174
26.2.5.1 IP Checksum 1174
26.2.5.2 TCP Checksum 1175
26.2.5.3 SCTP CRC Offloading 1175
26.2.5.4 Checksum Supported Per Packet Types 1176
26.2.6 Multiple Transmit Queues 1177
26.3 Interrupts 1177
26.3.1 Mapping of Interrupt Causes 1177
26.3.1.1 Legacy and MSI Interrupt Modes 1177
26.3.1.2 MSI-X Mode - VMDq Mode 1179
26.3.2 Legacy Interrupt Registers 1180
26.3.2.1 Interrupt Cause Register (ICR) 1181
26.3.2.2 Interrupt Cause Set Register (ICS) 1181
26.3.2.3 Interrupt Mask Set/Read Register (IMS) 1181
26.3.2.4 Interrupt Mask Clear Register (IMC) 1181
26.3.2.5 Interrupt Acknowledge Auto-Mask Register (IAM) 1182
26.3.2.6 Extended Interrupt Cause Registers (EICR) 1182
26.3.2.7 Extended Interrupt Cause Set Register (EICS) 1182
26.3.2.8 Extended Interrupt Mask Set and Read Register (EIMS) & Extended Interrupt Mask Clear Register (EIMC) 1182
26.3.2.9 Extended Interrupt Auto Clear Enable Register (EIAC) 1183
26.3.2.10 Extended Interrupt Auto Mask Enable Register (EIAM) 1183
26.3.2.11 GPIE Register 1183
26.3.3 MSI-X and Vectors 1184
26.3.4 Interrupt Moderation 1184
26.3.5 Clearing Interrupt Causes 1186
26.3.5.1 Auto-Clear 1186
26.3.5.2 Write to Clear 1186
26.3.5.3 Read to Clear 1187
26.3.6 Rate Controlled Low Latency Interrupts (LLI) 1187
26.3.6.1 Rate Control Mechanism 1187
26.3.7 TCP Timer Interrupt 1188
26.3.7.1 Introduction 1188
26.3.7.2 Description 1188
26.3.8 Special Cases About Interrupts 1189
26.3.8.1 Ordering Issue Between CSR Rd/Wr Received With Error by the Controller in the PCH and its Generated Interrupt 1189
26.4 802.1q VLAN Support 1189
26.4.1 802.1q VLAN Packet Format 1189
26.4.2 802.1q Tagged Frames 1189
26.4.3 Transmitting and Receiving 802.1q Packets 1190
26.4.3.1 Adding 802.1q Tags on Transmits 1190
26.4.3.2 Stripping 802.1q Tags on Receives 1190
26.4.4 802.1q VLAN Packet Filtering 1190
26.4.5 Double VLAN Support 1191
26.4.5.1 Transmit Behavior With External VLAN 1192
26.4.5.2 Receive Behavior With External VLAN 1192
26.5 Configurable LED Outputs 1193
26.5.1 MODE Encoding for LED Outputs 1193

26.6 Error Correction and Detection 1194
26.7 CPU Affinity Features 1194
26.7.1 Direct Cache Access (DCA) 1194
26.7.1.1 DCA Description 1194
26.7.1.2 Details of Implementation 1195
26.8 Virtualization 1196
26.8.1 Assignment of MSI-X Vectors to VM 1196
26.8.2 VM Resource Summary 1197
26.8.3 Packet Switching (VMDq) Model 1197
26.8.3.1 VMDq Assumptions 1197
26.8.3.2 VM Selection 1197
26.8.3.3 L2 Filtering 1197
26.8.3.4 VMDq Receive Packets Switching 1197
26.8.3.5 Mirroring Support 1202
26.8.3.6 VMDq Offload Support 1202
26.8.3.7 Security Features 1203
26.8.3.8 External Switch Loopback Support 1205
26.8.3.9 Switch Control 1206
26.8.4 Virtualization of the Hardware 1206
26.8.4.1 Per Pool Statistics 1206
26.9 Time SYNC (IEEE1588 and IEEE 802.1AS) 1207
26.9.1 Flow and Hardware/Software Responsibilities 1207
26.9.1.1 TimeSync Indications in Receive and Transmit Packet Descriptors 1209
26.9.2 Hardware Time Sync Elements 1209
26.9.2.1 System Time Structure and Mode of Operation 1209
26.9.2.2 Time Stamp Mechanism 1210
26.9.2.3 Time Adjustment Mode of Operation 1211
26.9.3 Time Sync Related Auxiliary Elements 1211
26.9.3.1 Target Time 1211
26.9.3.2 Configurable Frequency Clock 1212
26.9.3.3 Time Stamp Events 1213
26.9.4 Time SYNC Interrupts 1213
26.9.5 PTP Packet Structure 1213
26.10 Statistic Counters 1216
26.10.1 IEEE 802.3 Clause 30 Management 1216
26.10.2 OID_GEN_STATISTICS 1217
26.10.3 RMON 1218
26.10.4 Linux net_device_stats 1219

27.0 GbE Platform Manageability 1221
27.1 Platform Configurations 1221
27.1.1 On-Board BMC Configurations 1221
27.1.2 GbE Controller 1222
27.2 Pass Through Functionality 1222
27.2.1 SMBus Pass Through (PT) Functionality 1222
27.2.1.1 Pass Through (PT) Modes 1222
27.3 Programming Interfaces 1222
27.3.1 SMBus Programming 1222
27.3.1.1 Write SMBus Transactions (BMC to GbE Controller) 1223
27.3.1.2 Receive Enable Command 1224
27.3.1.3 Read SMBus Transactions (Controller to BMC) 1228
27.3.1.4 SMBus ARP Transactions 1237
27.4 Manageability Receive Filtering 1240
27.4.1 Overview and General Structure 1240
27.4.2 L2 Filters 1241
27.4.2.1 MAC and VLAN Filters 1241
27.4.2.2 EtherType Filters 1241

27.4.3 L3 and L4 Filters 1242
27.4.3.1 ARP Filtering 1242
27.4.3.2 Neighbor Discovery Filtering....1242
27.4.3.3 RMCP Port Filtering 1242
27.4.3.4 Flex Port Filtering....1242
27.4.3.5 The Controller IP Address Filtering 1242
27.4.3.6 Checksum Filter....1242
27.4.4 Manageability Decision Filters 1242
27.4.4.1 Exclusive Traffic 1245
27.4.5 Possible Configurations 1245
27.4.5.1 Dedicated MAC Packet Filtering....1245
27.4.5.2 Broadcast Packet Filtering....1245
27.4.5.3 VLAN Packet Filtering 1246
27.4.5.4 IPv6 Filtering 1246
27.4.5.5 Receive Filtering with Shared IP - CPMP....1246

28.0 GbE Programming Interface 1247
28.1 Introduction 1247
28.1.1 Memory, I/O Address, and Configuration Decoding 1247
28.1.1.1 Memory-Mapped Access to Internal Registers and Memories....1247
28.1.1.2 Memory-Mapped Access to MSI-X Tables....1247
28.1.1.3 I/O-Mapped Access to Internal Registers and Memories 1247
28.1.1.4 Configuration Access to Internal Registers and Memories 1249
28.1.2 Register Conventions....1250
28.1.2.1 Registers Byte Ordering....1251
28.1.3 Detailed Register Summary 1252
28.1.3.1 PCI Views 1252
28.1.4 Alias Addresses....1263
28.1.4.1 MSI-X BAR Register Summary 1265
28.1.4.2 Device Control Register—CTRL [0:3] (0x00000; R/W)....1265
28.1.4.3 Device Status Register—STATUS [0:3] (0x0008; R)....1269
28.1.4.4 Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W)....1270
28.1.4.5 MDI Control Register—MDIC [0:3] (0x0020; R/W)....1273
28.1.4.6 MDC/MDIO Configuration Register—MDICNFG [0:3] (0x0E04; R/W)....1274
28.1.4.7 Copper/Fiber Switch Control—CONNSW [0:3] (0x0034; R/W) 1275
28.1.4.8 VLAN Ether Type—VET [0:3] (0x0038; R/W) 1275
28.1.4.9 LED Control—LEDCTL [0:3] (0x0E00; RW) 1276
28.2 Internal Packet Buffer Size Registers....1276
28.2.1 Detailed Register Descriptions....1277
28.2.1.1 Internal Receive Packet Buffer Size—IRPBS [0:3](0x2404; RO)....1277
28.2.1.2 Internal Transmit Packet Buffer Size—ITPBS [0:3] (0x3404; RO) 1278
28.3 EEPROM Registers 1278
28.3.1 Detailed Register Descriptions....1278
28.3.1.1 EEPROM Control and Data Register—EEC [0:3] (0x0010; R/W) 1278
28.3.1.2 EEPROM Read Register - EERD [0:3] (0x0014; RW)....1280
28.3.1.3 EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO) 1281
28.3.1.4 VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO)....1282
28.3.1.5 Management—EEPROM CSR I/F 1283
28.3.1.6 Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO)....1283
28.3.1.7 Management EEPROM Read/Write Data—EEMNGDATA [0:3] (0x1014; RO)....1284
28.4 Flow Control Registers 1285
28.4.1 Detailed Register Descriptions....1285
28.4.1.1 Flow Control Address Low—FCAL [0:3] (0x0028; RO) 1285
28.4.1.2 Flow Control Address High—FCAH [0:3] (0x002C; RO)....1286
28.4.1.3 Flow Control Type—FCT [0:3] (0x0030; R/W)....1286
28.4.1.4 Flow Control Transmit Timer Value—FCTTV [0:3] (0x0170; R/W) 1287

28.4.1.5 Flow Control Receive Threshold Low—FCRTL0 [0:3] (0x2160; R/W) 1288
28.4.1.6 Flow Control Receive Threshold High—FCRTH0 [0:3] (0x2168; R/W).... 1289
28.4.1.7 Flow Control Refresh Threshold Value—FCRTV[0:3] (0x2460; R/W)..... 1290
28.4.1.8 Flow Control Status—FCSTS0 [0:3] (0x2464; RO) 1291
28.5 GbE PCIe* Registers 1291
28.5.1 Detailed Register Description 1291
28.5.1.1 Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO) . 1291
28.5.1.2 Mirrored Revision ID—MREVID[0:3] (0x5B64; R/W)..... 1293
28.5.1.3 PCIe* Control Extended Register—GCR_EXT[0:3] (0x5B6C; R/W)..... 1294
28.6 Semaphore Registers 1295
28.6.1 Detailed Register Descriptions 1295
28.6.1.1 Software Semaphore—SWSM[0:3] (0x5B50; R/W) 1295
28.6.1.2 Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS) 1296
28.6.1.3 Software-Firmware Synchronization—SW_FW_SYNC[0:3] (0x5B5C; RWS) Check with LAD for Phy Related Bits..... 1298
28.6.1.4 Software Mailbox Write—SWMBWR[0:3] (0x5B04; R/W)..... 1299
28.6.1.5 Software Mailbox 0—SWMB0[0:3] (0x5B08; RO) 1299
28.6.1.6 Software Mailbox 1—SWMB1[0:3] (0x5B0C; RO)..... 1299
28.6.1.7 Software Mailbox 2—SWMB2[0:3] (0x5B18; RO) 1300
28.6.1.8 Software Mailbox 3—SWMB3[0:3] (0x5B1C; RO)..... 1300
28.7 Interrupt Register Descriptions 1301
28.7.1 Detailed Register Descriptions 1301
28.7.1.1 Extended Interrupt Cause—EICR[0:3] (0x1580; RC/W1C) 1301
28.7.1.2 EICR Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1301
28.7.1.3 EICR Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1).. 1302
28.7.1.4 Extended Interrupt Cause Set—EICS [0:3] (0x1520; WO) 1302
28.7.1.5 EICS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1302
28.7.1.6 EICS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1).. 1303
28.7.1.7 Extended Interrupt Mask Set/Read—EIMS [0:3] (0x1524; RWS)..... 1303
28.7.1.8 EIMS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1303
28.7.1.9 EIMS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) . 1304
28.7.1.10 Extended Interrupt Mask Clear - EIMC [0:3] (0x1528; WO) 1304
28.7.1.11 EIMC Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1305
28.7.1.12 EIMC Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) . 1305
28.7.1.13 Extended Interrupt Auto Clear—EIAC [0:3] (0x152C; R/W)..... 1306
28.7.1.14 Extended Interrupt Auto Mask Enable—EIAM [0:3] (0x1530; R/W) 1307
28.7.1.15 EIAM Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1307
28.7.1.16 EIAM Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) . 1307
28.7.1.17 Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C)..... 1308
28.7.1.18 Interrupt Cause Set Register—ICS [0:3] (0x1504; WO)..... 1310
28.7.1.19 Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W)..... 1312
28.7.1.20 Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO)..... 1314
28.7.1.21 Interrupt Acknowledge Auto Mask Register—IAM [0:3] (0x1510; R/W) 1316
28.7.1.22 Interrupt Throttle—EITR [0:3] (0x1680 + 4*n [n = 0...9]; R/W) 1316
28.7.1.23 Interrupt Vector Allocation Registers—IVAR [0:3] (0x1700 + 4*n [n=0...3]; RW) 1318
28.7.1.24 Interrupt Vector Allocation Registers—MISC IVAR_MISC [0:3] (0x1740; RW)..... 1319
28.7.1.25 General Purpose Interrupt Enable—GPIE [0:3] (0x1514; RW)..... 1320
28.8 MSI-X Table Register Descriptions..... 1320
28.8.1 Detailed Register Descriptions 1321
28.8.1.1 MSIX Table Entry Lower Address—MSIXTADD [0:3][0:9] (BAR3: 0x0000 + 0x10*n [n=0...9]; R/W) 1321

28.8.1.2 MSIX Table Entry Upper Address—MSIXTUADD [0:3][0:9] (BAR3: 0x0004 + 0x10*n [n=0...9]; R/W) 1322
28.8.1.3 MSIX Table Entry Message—MSIXTMSG [0:3][0:9] (BAR3: 0x0008 + 0x10*n [n=0...9]; R/W) 1322
28.8.1.4 MSIX Table Entry Vector Control—MSIXTVCTRL [0:3][0:9] (BAR3: 0x000C + 0x10*n [n=0...9]; R/W) 1323
28.8.1.5 MSIXPBA Bit Description—MSIXPBA [0:3] (BAR3: 0x2000; RO) 1323
28.8.1.6 MSIX PBA Clear—PBACL [0:3] (0x5B68; R/W1C) 1324
28.9 Receive Registers 1325
28.9.1 Detailed Register Descriptions 1325
28.9.1.1 Receive Control Register—RCTL [0:3] (0x0100; R/W) 1325
28.9.1.2 Split and Replication Receive Control—SRRCTL [0:3][0:7] (0xC00C + 0x40*n [n=0...7]; R/W) 1328
28.9.1.3 Packet Split Receive Type—PSRTYPE [0:3][0:7] (0x5480 + 4*n [n=0...7]; R/W) 1331
28.9.1.4 Replicated Packet Split Receive Type—RPLPSRTYPE [0:3] (0x54C0; R/W) 1332
28.9.1.5 Receive Descriptor Base Address Low—RDBAL [0:3] [0:7] (0xC000 + 0x40*n [n=0...7]; R/W) 1333
28.9.1.6 Receive Descriptor Base Address High—RDBAH [0:3] [0:7] (0xC004 + 0x40*n [n=0...7]; R/W) 1333
28.9.1.7 Receive Descriptor Ring Length—RDLEN [0:3][0:7] (0xC008 + 0x40*n [n=0...7]; R/W) 1334
28.9.1.8 Receive Descriptor Head—RDH [0:3][0:7] (0xC010 + 0x40*n [n=0...7]; RO) 1334
28.9.1.9 Receive Descriptor Tail—RDT [0:3][0:7] (0xC018 + 0x40*n [n=0...7]; R/W) 1335
28.9.1.10 Receive Descriptor Control—RXDCTL [0:3][0:7] (0xC028 + 0x40*n [n=0...7]; R/W) 1335
28.9.1.11 Receive Queue Drop Packet Count—RQDPC [0:3][0:7] (0xC030 + 0x40*n [n=0...7]; RC/W) 1337
28.9.1.12 Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W) 1338
28.9.1.13 Receive Long Packet Maximum Length—RLPML [0:3] (0x5004; R/W) 1340
28.9.1.14 Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W) 1340
28.9.1.15 Multicast Table Array—MTA [0:3][0:127] (0x5200 + 4*n [n=0...127]; R/W) 1341
28.9.1.16 Receive Address Low 0—RAL0 [0:3][0:15] (0x5400 + 8*n [n=0...15]; R/W) 1342
28.9.1.17 Receive Address Low 1—RAL1 [0:3][0:7] (0x54E0 + 8*n [n=0...7]; R/W) 1343
28.9.1.18 Receive Address High 0—RAH0 [0:3][0:15] (0x5404 + 8*n [n=0...15]; R/W) 1343
28.9.1.19 Receive Address High 1—RAH1 [0:3][0:7] (0x54E4 + 8*n [n=0...7]; R/W) 1345
28.9.1.20 VLAN Filter Table Array—VFTA [0:3][0:127] (0x5600 + 4*n [n=0...127]; R/W) 1346
28.9.1.21 Multiple Receive Queues Command Register—MRQC [0:3] (0x5818; R/W) 1347
28.9.1.22 RSS Random Key Register—RSSRK [0:3][0:9] (0x5C80 + 4*n [n=0...9]; R/W) 1349
28.9.1.23 Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W) 1349
28.10 Filtering Register 1351
28.10.1 Detailed Register Descriptions 1351
28.10.1.1 Immediate Interrupt RX—IMIR [0:3][0:7] (0x5A80 + 4*n [n=0...7]; R/W) 1351

28.10.1.2 Immediate Interrupt Rx Ext.—IMIREXT [0:3][0:7] (0x5AA0 + 4*n [n=0...7]; R/W) 1352
28.10.1.3 2tuples Queue Filter—TTQF [0:3][0:7] (0x59E0 + 4*n[n=0..7];RW) 1353
28.10.1.4 Immediate Interrupt Rx VLAN Priority—IMIRVP [0:3] (0x5AC0; R/W) ... 1354
28.10.1.5 SYN Packet Queue Filter—SYNQF [0:3] (0x55FC; RW) 1354
28.10.1.6 EType Queue Filter—ETQF [0:3][0:7] (0x5CB0 + 4*n[n=0...7]; R/W) .. 1355
28.11 Transmit Registers 1356
28.11.1 Detailed Register Descriptions 1356
28.11.1.1 Transmit Control Register—TCTL [0:3] (0x0400; R/W) 1356
28.11.1.2 Transmit Control Extended—TCTL_EXT [0:3] (0x0404; R/W) 1357
28.11.1.3 Transmit IPG Register—TIPG (0x0410; R/W) 1358
28.11.1.4 Retry Buffer Control—RETX_CTL [0:3] (0x041C; R/W) 1359
28.11.1.5 DMA Tx Control—DTXCTL [0:3] (0x3590; R/W) 1359
28.11.1.6 DMA TX TCP Flags Control Low—DTXTCPFLGL [0:3] (0x359C; R/W) 1360
28.11.1.7 DMA TX TCP Flags Control High—DTXTCPFLGH [0:3] (0x35A0; RW) 1361
28.11.1.8 DMA TX Max Total Allow Size Requests—DTXMXSZRQ [0:3] (0x3540; RW) 1361
28.11.1.9 DMA TX Maximum Packet Size—DTXMXPKTSZ [0:3] (0x355C; RW) 1362
28.11.1.10 Transmit Descriptor Base Address Low—TDBAL [0:3][0:7] (0xE000 + 0x40*n [n=0...7]; R/W) 1362
28.11.1.11 Transmit Descriptor Base Address High—TDBAH [0:3][0:7] (0xE004 + 0x40*n [n=0...7]; R/W) 1363
28.11.1.12 Transmit Descriptor Ring Length—TDLEN [0:3][0:7] (0xE008 + 0x40*n [n=0...7]; R/W) 1363
28.11.1.13 Transmit Descriptor Head—TDH [0:3][0:7] (0xE010 + 0x40*n [n=0...7]; RO) 1364
28.11.1.14 Transmit Descriptor Tail—TDT [0:3][0:7] (0xE018 + 0x40*n [n=0...7]; R/W) 1364
28.11.1.15 Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W) 1365
28.11.1.16 Tx Descriptor Completion Write-Back Address Low—TDWBAL [0:3] [0:7] (0xE038 + 0x40*n [n=0...7]; R/W) 1367
28.11.1.17 Tx Descriptor Completion Write-Back Address High - TDWBAH [0:3] [0:7] (0xE03C + 0x40*n [n=0...7];R/W) 1367
28.12 DCA and TPH Registers 1368
28.12.1 Detailed Register Descriptions 1368
28.12.1.1 Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W) 1368
28.12.1.2 Tx DCA Control Registers—TXCTL [0:3][0:7] (0xE014 + 0x40*n [n=0...7]; R/W) 1370
28.12.1.3 DCA Requester ID Information—DCA_ID [0:3] (0x5B70; RO) 1371
28.12.1.4 DCA Control—DCA_CTRL [0:3] (0x5B74; R/W) 1371
28.13 Virtualization Registers 1372
28.13.1 Detailed Register Descriptions 1372
28.13.1.1 VMDq Control Register—VT_CTL [0:3] (0x581C; R/W) 1372
28.13.1.2 Malicious Driver Free Block—MDFB [0:3] (0x3558; RWS) 1373
28.13.1.3 Last VM Misbehavior Cause—LVMMC [0:3] (0x3548; RC) 1373
28.13.1.4 VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW) 1374
28.13.1.5 Replication Offload Register—RPLOLR [0:3] (0x5AF0; RW) 1375
28.13.1.6 VLAN VM Filter—VLVF [0:3][0:31] (0x5D00 + 4*n [n=0...31]; RW) 1376
28.13.1.7 Unicast Table Array—UTA [0:3][0:127] (0xA000 + 4*n [n=0...127]; R/W) 1377
28.13.1.8 Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW) 1377
28.13.1.9 Storm Control Status—SCSTS [0:3] (0x5DB4;RO) 1378
28.13.1.10 Broadcast Storm Control Threshold—BSCTRH [0:3] (0x5DB8;RW) 1379
28.13.1.11 Multicast Storm Control Threshold—MSCTRH [0:3] (0x5DBC; RW) 1379
28.13.1.12 Broadcast Storm Control Current Count - BSCCNT [0:3] (0x5DC0;RO) . 1379
28.13.1.13 Multicast Storm Control Current Count—MSCCNT [0:3] (0x5DC4;RO) ... 1380

28.13.1.14 Storm Control Time Counter—SCTC [0:3] (0x5DC8; RO).....1380
28.13.1.15 Storm Control Basic Interval—SCBI [0:3] (0x5DCC; RW) 1381
28.13.1.16 Virtual Mirror Rule Control—VMRCTL [0:3][0:3] (0x5D80 + 0x4*n [n= 0...3]; RW).....1381
28.13.1.17 Virtual Mirror Rule VLAN—VMRVLAN [0:3][0:3] (0x5D90 + 0x4*n [n= 0...3]; RW).....1382
28.13.1.18 Virtual Mirror Rule VM—VMRVM [0:3][0:3] (0x5DA0 + 0x4*n [n= 0...3]; RW).....1382
28.14 Power Management Registers 1383
28.14.1 Detailed Register Descriptions.....1383
28.14.1.1 DMA Receive Power Saving Register—DMARPS [0:3] (0x2500; R/W) 1383
28.14.1.2 DMA Transmit Power Saving Register—DMATPS [0:3] (0x2504; R/W) ...1384
28.15 Timer Registers.....1385
28.15.1 Detailed Register Descriptions.....1385
28.15.1.1 Watchdog Setup—WDSTP [0:3] (0x1040; R/W) 1385
28.15.1.2 Watchdog Software Device Status—WDSWSTS [0:3] (0x1044; R/W) 1386
28.15.1.3 Free Running Timer—FRTIMER [0:3] (0x1048; RWS).....1386
28.15.1.4 TCP Timer—TCPTIMER [0:3] (0x104C; R/W).....1387
28.16 Time Sync Register Descriptions.....1388
28.16.1 Detailed Register Descriptions.....1388
28.16.1.1 RX Time Sync Control Register—TSYNCRXCTL [0:3] (0xB620;RW) 1388
28.16.1.2 RX timestamp Low—RXSTMPL [0:3] (0xB624; RO).....1389
28.16.1.3 RX Timestamp High—RXSTMPH [0:3] (0xB628; RO).....1389
28.16.1.4 RX Timestamp Attributes Low—RXSATRL[0:3] (0xB62C; RO).....1389
28.16.1.5 RX timestamp Attributes High—RXSATRH [0:3] (0xB630; RO) 1390
28.16.1.6 TX Time Sync Control Register—TSYNCTXCTL [0:3] (0xB614; RW).....1390
28.16.1.7 TX Timestamp Value Low—TXSTMPL [0:3] (0xB618;RO) 1391
28.16.1.8 TX Timestamp Value High—TXSTMPH[0:3] (0xB61C; RO).....1391
28.16.1.9 System Time Register Residue—SYSTIMR [0:3] (0xB6F8; RW).....1391
28.16.1.10 System Time Register Low—SYSTIML [0:3] (0xB600; RW) 1392
28.16.1.11 System Time Register High—SYSTIMH [0:3] (0xB604; RW).....1392
28.16.1.12 Increment Attributes Register—TIMINCA [0:3] (0xB608; RW).....1393
28.16.1.13 Time Adjustment Offset Register Low—TIMADJL [0:3] (0xB60C; RW) ...1393
28.16.1.14 Time Adjustment Offset Register High—TIMADJH [0:3] (0xB610;RW) ...1393
28.16.1.15 TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW) 1394
28.16.1.16 Target Time Register 0 Low—TRGTTIML0 [0:3] (0xB644; RW) 1396
28.16.1.17 Target Time Register 0 High—TRGTTIMH0 [0:3] (0xB648; RW).....1396
28.16.1.18 Target Time Register 1 Low—TRGTTIML1 [0:3] (0xB64C; RW).....1396
28.16.1.19 Target Time Register 1 High—TRGTTIMH1 [0:3] (0xB650; RW).....1397
28.16.1.20 Frequency Out 0 Control Register—FREQOUT0 [0:3] (0xB654; RW).....1397
28.16.1.21 Frequency Out 1 Control Register—FREQOUT1 [0:3] (0xB658; RW).....1398
28.16.1.22 Auxiliary Time Stamp 0 Register Low—AUXSTMPL0 [0:3] (0xB65C; RO) 1398
28.16.1.23 Auxiliary Time Stamp 0 Register High—AUXSTMPH0 [0:3] (0xB660; RO)..... 1399
28.16.1.24 Auxiliary Time Stamp 1 Register Low—AUXSTMPL1 [0:3] (0xB664; RO) 1399
28.16.1.25 Auxiliary Time Stamp 1 Register High—AUXSTMPH1 [0:3] (0xB668; RO)..... 1400
28.16.1.26 Time Sync RX Configuration—TSYNCRXCFG [0:3] (0x5F50; R/W).....1400
28.16.1.27 Time Sync SDP Configuration Register—TSSDP [0:3] (0x003C; R/W)....1401
28.16.2 Time Sync Interrupt Registers.....1402
28.16.2.1 Time Sync Interrupt Cause Register—TSICR [0:3] (0xB66C; RC/W1C) ..1402
28.16.2.2 Time Sync Interrupt Mask Register—TSIM [0:3] (0xB674; RW).....1403
28.16.2.3 Time Sync Interrupt Set Register—TSIS [0:3] (0xB670; WO) 1404
28.17 PCS Registers 1405
28.17.1 Detailed Register Descriptions.....1405
28.17.1.1 PCS Configuration—PCS_CFG [0:3] (0x4200; R/W) 1405
28.17.1.2 PCS Link Control—PCS_LCTL [0:3] (0x4208; RW) 1405
28.17.1.3 PCS Link Status—PCS_LSTS [0:3] (0x420C; RO) 1407
28.17.1.4 AN Advertisement—PCS_ANADV [0:3] (0x4218; R/W).....1409

28.17.1.5 Link Partner Ability—PCS_LPAB [0:3] (0x421C; RO) 1410
28.17.1.6 Next Page Transmit—PCS_NPTX [0:3] (0x4220; RW) 1411
28.17.1.7 Link Partner Ability Next Page—PCS_LPABNP [0:3] (0x4224; RO) 1412
28.17.1.8 SFP I2C Command—I2CCMD [0:3] (0x1028; R/W) 1413
28.17.1.9 SFP I2C Parameters—I2CPARAMS [0:3] (0x102C; R/W) 1414
28.18 Statistics Register Descriptions 1415
28.18.1 Detailed Register Descriptions 1415
28.18.1.1 CRC Error Count—CRCERRS [0:3] (0x4000; RC) 1415
28.18.1.2 Alignment Error Count—ALGNERRC [0:3] (0x4004; RC) 1416
28.18.1.3 Symbol Error Count—SYMERRS [0:3] (0x4008; RC) 1416
28.18.1.4 Missed Packets Count—MPC [0:3] (0x4010; RC) 1417
28.18.1.5 Single Collision Count—SCC [0:3] (0x4014; RC) 1417
28.18.1.6 Excessive Collisions Count—ECOL [0:3] (0x4018; RC) 1418
28.18.1.7 Multiple Collision Count—MCC [0:3] (0x401C; RC) 1418
28.18.1.8 Late Collisions Count—LATECOL [0:3] (0x4020; RC) 1419
28.18.1.9 Collision Count—COLC [0:3] (0x4028; RC) 1419
28.18.1.10 Defer Count—DC [0:3] (0x4030; RC) 1420
28.18.1.11 Host Transmit Discarded Packets by MAC Count—HTDPMC [0:3] (0x403C; RC) 1420
28.18.1.12 Receive Length Error Count—RLEC [0:3] (0x4040; RC) 1421
28.18.1.13 XON Received Count—XONRXC [0:3] (0x4048; RC) 1421
28.18.1.14 XON Transmitted Count—XONTXC [0:3] (0x404C; RC) 1422
28.18.1.15 XOFF Received Count—XOFFRXC [0:3] (0x4050; RC) 1422
28.18.1.16 XOFF Transmitted Count—XOFFTXC [0:3] (0x4054; RC) 1423
28.18.1.17 FC Received Unsupported Count—FCRUC [0:3] (0x4058; RC) 1423
28.18.1.18 Packets Received [64 Bytes] Count—PRC64 [0:3] (0x405C; RC) 1424
28.18.1.19 Packets Received [65-127 Bytes] Count—PRC127 [0:3] (0x4060; RC) .. 1424
28.18.1.20 Packets Received [128-255 Bytes] Count—PRC255 [0:3] (0x4064; RC) 1425
28.18.1.21 Packets Received [256-511 Bytes] Count—PRC511 [0:3] (0x4068; RC) 1426
28.18.1.22 Packets Received [512-1023 Bytes] Count—PRC1023 [0:3] (0x406C; RC) 1426
28.18.1.23 Packets Received [1024 to Max Bytes] Count—PRC1522 [0:3] (0x4070; RC) 1427
28.18.1.24 Good Packets Received Count—GPRC [0:3] (0x4074; RC) 1427
28.18.1.25 Broadcast Packets Received Count—BPRC [0:3] (0x4078; RC) 1428
28.18.1.26 Multicast Packets Received Count—MPRC [0:3] (0x407C; RC) 1428
28.18.1.27 Good Packets Transmitted Count—GPTC [0:3] (0x4080; RC) 1429
28.18.1.28 Good Octets Received Count—GORCL [0:3] (0x4088; RC) 1429
28.18.1.29 Good Octets Received Count—GORCH [0:3] (0x408C; RC) 1430
28.18.1.30 Good Octets Transmitted Count—GOTCL [0:3] (0x4090; RC) 1430
28.18.1.31 Good Octets Transmitted Count—GOTCH [0:3] (4094; RC) 1431
28.18.1.32 Receive No Buffers Count—RNBC [0:3] (0x40A0; RC) 1431
28.18.1.33 Receive Undersize Count—RUC [0:3] (0x40A4; RC) 1432
28.18.1.34 Receive Fragment Count—RFC [0:3] (0x40A8; RC) 1432
28.18.1.35 Receive Oversize Count—ROC [0:3] (0x40AC; RC) 1433
28.18.1.36 Receive Jabber Count—RJC [0:3] (0x40B0; RC) 1433
28.18.1.37 Management Packets Received Count—MNGPRC [0:3] (0x40B4; RC) 1434
28.18.1.38 Management Packets Dropped Count—MPDC [0:3] (0x40B8; RC) 1434
28.18.1.39 Management Packets Transmitted Count—MNGPTC [0:3] (0x40BC; RC) 1435
28.18.1.40 Total Octets Received—TORL [0:3] (0x40C0; RC) 1435
28.18.1.41 Total Octets Received—TORH [0:3] (0x40C4; RC) 1436
28.18.1.42 Total Octets Transmitted—TOTL [0:3] (0x40C8; RC) 1436
28.18.1.43 Total Octets Transmitted—TOTH [0:3] (0x40CC; RC) 1437
28.18.1.44 Total Packets Received—TPR [0:3] (0x40D0; RC) 1437
28.18.1.45 Total Packets Transmitted—TPT [0:3] (0x40D4; RC) 1438
28.18.1.46 Packets Transmitted [64 Bytes] Count—PTC64 [0:3] (0x40D8; RC) 1438

28.18.1.47 Packets Transmitted [65-127 Bytes] Count—PTC127 [0:3] (0x40DC; RC)....1439
28.18.1.48 Packets Transmitted [128-255 Bytes] Count—PTC255 [0:3] (0x40E0; RC)....1439
28.18.1.49 Packets Transmitted [256-511 Bytes] Count—PTC511 [0:3] (0x40E4; RC)....1440
28.18.1.50 Packets Transmitted [512-1023 Bytes] Count—PTC1023 [0:3] (0x40E8; RC)....1440
28.18.1.51 Packets Transmitted [1024 Bytes or Greater] Count—PTC1522 [0:3] (0x40EC; RC)....1441
28.18.1.52 Multicast Packets Transmitted Count—MPTC [0:3] (0x40F0; RC) 1441
28.18.1.53 Broadcast Packets Transmitted Count—BPTC [0:3] (0x40F4; RC) 1442
28.18.1.54 TCP Segmentation Context Transmitted Count—TSCTC [0:3] (0x40F8; RC)....1442
28.18.1.55 Interrupt Assertion Count—IAC [0:3] (0x4100; RC)....1443
28.18.1.56 Rx Packets to Host Count—RPTHC [0:3] (0x4104; RC)....1443
28.18.1.57 Debug Counter 1—DBGC1 [0:3] (0x4108; RC)....1443
28.18.1.58 Debug Counter 2—DBGC2 [0:3] (0x410C; RC) 1444
28.18.1.59 Debug Counter 3—DBGC3 [0:3] (0x4110; RC)....1445
28.18.1.60 Debug Counter 4—DBGC4 [0:3] (0x411C; RC) 1445
28.18.1.61 Host Good Packets Transmitted Count—HGPTC [0:3] (0x4118; RC) 1446
28.18.1.62 Receive Descriptor Minimum Threshold Count—RXDMTC [0:3] (0x4120; RC)....1446
28.18.1.63 Host Good Octets Received Count—HGORCL [0:3] (0x4128; RC)....1447
28.18.1.64 Host Good Octets Received Count—HGORCH [0:3] (0x412C; RC) 1447
28.18.1.65 Host Good Octets Transmitted Count—HGOTCL [0:3] (0x4130; RC)....1448
28.18.1.66 Host Good Octets Transmitted Count - HGOTCH [0:3] (0x4134; RC)....1448
28.18.1.67 Length Error Count—LENERRS [0:3] (0x4138; RC)....1449
28.18.1.68 SerDes/SGMII/KX Code Violation Packet Count—SCVPC [0:3] (0x4228; RW)....1449
28.18.1.69 Switch Drop Packet Count—SDPC [0:3] (0x41A4; RC) 1450
28.18.1.70 Virtualization Statistical Counters....1450
28.18.1.71 Per Queue Good Packets Received Count—VFGPRC [0:3][0:7] (0x10010
+ n*0x100 [n=0...7]; RO)....1450
28.18.1.72 Per Queue Good Packets Transmitted Count—VFGPTC [0:3][0:7] (0x10014 + n*0x100 [n=0...7]; RO)....1451
28.18.1.73 Per Queue Good Octets Received Count—VFGORC [0:3][0:7] (0x10018 + n*0x100 [n=0...7]; RO)....1452
28.18.1.74 Per Queue Good Octets Transmitted Count—VFGOTC [0:3][0:7] (0x10034 + n*0x100 [n=0...7]; RO)....1453
28.18.1.75 Per Queue Multicast Packets Received Count—VFMPRC [0:3][0:7] (0x10038 + n*0x100 [n=0...7]; RO)....1453
28.19 Manageability Statistics....1454
28.19.1 Detailed Register Descriptions....1454
28.19.1.1 BMC Management Packets Dropped Count—BMPDC [0:3] (0x4140; RC)....1454
28.19.1.2 BMC Management Packets Transmitted Count—BMNGPTC [0:3] (0x4144; RC)....1454
28.19.1.3 BMC Management Packets Received Count—BMNGPRC [0:3] (0x413C; RC)....1455
28.19.1.4 BMC Total Unicast Packets Received—BUPRC [0:3] (0x4400; RC)....1455
28.19.1.5 BMC Total Multicast Packets Received—BMPRC [0:3] (0x4404; RC) 1456
28.19.1.6 BMC Total Broadcast Packets Received—BBPRC [0:3] (0x4408; RC) 1456
28.19.1.7 BMC Total Unicast Packets Transmitted—BUPTC [0:3] (0x440C; RC)....1456
28.19.1.8 BMC Total Multicast Packets Transmitted—BMPTC [0:3] (0x4410; RC) ..1457
28.19.1.9 BMC Total Broadcast Packets Transmitted—BBPTC [0:3] (0x4414; RC) .1457
28.19.1.10 BMC FCS Receive Errors—BCRCERRS [0:3] (0x4418; RC)....1458
28.19.1.11 BMC Alignment Errors—BALGNERRC [0:3] (0x441C; RC) 1458
28.19.1.12 BMC Pause XON Frames Received—BXONRXC [0:3] (0x4420; RC)....1458

28.19.1.13 BMC Pause XOFF Frames Received—BXOFFRXC [0:3] (0x4424; RC) 1459
28.19.1.14 BMC Pause XON Frames Transmitted—BXONTXC [0:3] (0x4428; RC) ... 1459
28.19.1.15 BMC Pause XOFF Frames Transmitted—BXOFFTXC [0:3] (0x442C; RC) 1460
28.19.1.16 BMC Single Collision Transmit Frames—BSCC [0:3] (0x4430; RC)..... 1460
28.19.1.17 BMC Multiple Collision Transmit Frames—BMCC [0:3] (0x4434; RC) 1460
28.20 Wake Up Controls Registers 1461
28.20.1 Detailed Register Descriptions 1461
28.20.1.1 Wakeup Control Register—WUC [0:3] (0x5800; R/W) 1461
28.20.1.2 Wakeup Filter Control Register—WUFC [0:3] (0x5808; R/W) 1462
28.20.1.3 Wakeup Status Register—WUS [0:3] (0x5810; R/W1C) 1463
28.20.1.4 Wakeup Packet Length—WUPL [0:3] (0x5900; RO)..... 1464
28.20.1.5 Wakeup Packet Memory—WUPM [0:3][0:31] (0x5A00 + 4*n [n=0...31]; RO) 1464
28.20.1.6 IP Address Valid—IPAV [0:3] (0x5838; R/W) 1465
28.20.1.7 IPv4 Address Table—IP4AT [0:3][0:3] (0x5840 + 8*n [n=0...3]; RW) . 1465
28.20.1.8 IPv6 Address Table—IP6AT [0:3][0:3] (0x5880 + 4*n [n=0...3]; RW) . 1466
28.20.2 Flexible Host Filter Table Registers—FHFT [0:3] (0x9000 - 0x93FC; RW)..... 1466
28.20.2.5 Flex Filter 0—Example 1470
28.20.3 Flexible Host Filter Table Extended Registers—FHFT_EXT [0:3] (0x9A00 - 0x9DFC; RW)..... 1470
28.21 Management Registers 1473
28.21.1 Detailed Register Descriptions 1473
28.21.1.1 Management VLAN TAG Value—MAVTV [0:3][0:7] (0x5010 +4*n [n=0...7]; RW) 1473
28.21.1.2 Management Flex UDP/TCP Ports—MFUTP[0:3][0:3] (0x5030 + 4*n [n=0...3]; RW) 1473
28.21.1.3 Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW) 1474
28.21.1.4 Management Control Register—MANC [0:3] (0x5820; RW) 1475
28.21.1.5 Management Only Traffic Register—MNGONLY [0:3] (0x5864; RW)..... 1476
28.21.1.6 Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW) 1476
28.21.1.7 Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW) 1478
28.21.1.8 Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW) 1478
28.21.1.9 Manageability MAC Address Low—MMAL [0:3] (0x5910 + 8*n [n= 0...1]; RW) 1481
28.21.1.10 Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0...1]; RW) 1482
28.21.1.11 Flexible TCO Filter Table Registers—FTFT [0:3] (0x9400-0x94FC; RW) 1482
28.22 Memory Error Registers Description..... 1484
28.22.1 Parity and ECC Error Indication—PEIND [0:3] (0x1084; RC) 1484
28.22.2 Parity and ECC Indication Mask—PEINDM [0:3] (0x1088; RW)..... 1485
28.22.3 DMA Transmit Descriptor Parity Status—DTPARS [0:3] (0x3510; RW1C)..... 1485
28.22.4 DMA Receive Descriptor Parity Status—DRPARS [0:3] (0x3514; RW1C) 1486
28.22.5 Dhost Parity Status—DDPARS [0:3] (0x3518; RW1C) 1486
28.22.6 Tx Packet Buffer ECC Status—TPBECCSTS [0:3] (0x345C; RW) 1487
28.22.7 LAN Port Parity Error Control Register—LANPERRCTL [0:3] (0x5F54; RW) 1487
28.22.8 LAN Port Parity Error Status Register—LANPERRSTS [0:3] (0x5F58; RO)..... 1488

29.0 Function 1-4 (GbE) 1489
29.1 Introduction..... 1489
29.2 Detailed Register Summary..... 1492
29.2.1 PCI Views..... 1492
29.3 Mandatory PCI Configuration Registers 1495
29.3.1 Detailed Register Descriptions 1495

29.3.1.1 PVID[0:3]—PF Vendor Identification Register....1495
29.3.1.2 PDID0—PF Device Identification Register (GbE0)....1496
29.3.1.3 PDID1—PF Device Identification Register (GbE1)....1496
29.3.1.4 PDID2—PF Device Identification Register (GbE2)....1497
29.3.1.5 PDID3—PF Device Identification Register (GbE3)....1497
29.3.1.6 PPCICMD[0:3]—PF Device Command Register....1498
29.3.1.7 PPCISTS[0:3]—PF Device Status Register....1499
29.3.1.8 PRID[0:3]—PF Revision ID Register....1501
29.3.1.9 PCC[0:3]—PF Class Code Register....1501
29.3.1.10 PHDR[0:3]—PF Header Type Register....1502
29.3.2 Base Address Registers (0x10...0x27; R/W)....1502
29.3.2.1 Base Address Register Fields....1502
29.3.2.2 GbEPCIBAR0[0:3]—BAR0 Base Address Register....1503
29.3.2.3 GbEPCIBAR1[0:3]—BAR1 Base Address Register....1504
29.3.2.4 GbEPCIBAR2[0:3]—BAR2 Base Address Register....1504
29.3.2.5 GbEPCIBAR3[0:3]—BAR3 Base Address Register....1505
29.3.2.6 GbEPCIBAR4[3:0]—BAR4 Base Address Register....1506
29.3.2.7 GbEPCIBAR5[0:3]—BAR5 Base Address Register....1507
29.3.2.8 PSVID[0:3]—PF Subsystem Vendor ID Register....1507
29.3.2.9 PSID[0:3]—PF Subsystem ID Register....1508
29.3.2.10 Expansion ROM Base Address (0x30; RO) (Flash Not Implemented)....1508
29.3.2.11 PCP[0:3]—PF Capabilities Pointer Register....1508
29.3.2.12 PIRQL[0:3]—PF Interrupt Line Register....1509
29.3.2.13 PIRQP0—PF Interrupt Pin Register 0....1509
29.3.2.14 PIRQP1—PF Interrupt Pin Register 1....1510
29.3.2.15 PIRQP2—PF Interrupt Pin Register 2....1510
29.3.2.16 PIRQP3—PF Interrupt Pin Register 3....1511
29.4 PCI Capabilities....1511
29.4.1 Power Management Capability Structure....1511
29.4.1.1 PPMCAP[0:3]—PF Power Management Capabilities ID Register....1512
29.4.1.2 PPMCP[0:3]—PF Power Management Next Capability Pointer Register....1512
29.4.1.3 PPMC[0:3]—PF Power Management Capabilities Register....1513
29.4.1.4 PPMCSR—PF Power Management Control and Status Register....1514
29.4.2 MSI Capability Structure....1515
29.4.2.1 PMSICID[0:3]—Message Signalled Interrupt Capability ID Register....1515
29.4.2.2 PMSINCP[0:3]—Message Signalled Interrupt Next Capability Pointer Register....1515
29.4.2.3 PMSICTL[0:3]—Message Signalled Interrupt Control Register....1516
29.4.2.4 PMSILADDR[0:3]—Message Signalled Interrupt Lower Address Register 1516
29.4.2.5 PMSIUADDR[0:3]—Message Signalled Interrupt Upper Address Register1517
29.4.2.6 PMSIDATA[0:3]—Message Signalled Interrupt Data Register....1517
29.4.2.7 PMSIMSK[0:3]—Message Signalled Interrupt Mask Register....1517
29.4.2.8 PMSIPND[0:3]—Message Signalled Interrupt Mask Pending Register....1518
29.4.3 MSI-X Configuration....1518
29.4.3.1 PMSI-X[0:3]—PF Message Signalled Interrupt X Capability ID Register..1519
29.4.3.2 PMSIXNCP[0:3]—PF MSIX Next Capability Pointer Register....1520
29.4.3.3 PMSIXCNTL[0:3]—PF Message Signalled Interrupt X Control Register....1520
29.4.3.4 PMSIXTBIR[0:3]—PF MSI-X Table Offset & Table BIR Register....1521
29.4.3.5 PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register .1521
29.4.4 CSR Access Via Configuration Address Space....1522
29.4.4.1 IOADDR[0:3]—IOADDR Register....1522
29.4.4.2 IODATA[0:3]—IODATA Register....1522
29.4.5 PCIe* Configuration Registers....1523
29.4.6 PCI Express Capability Structure....1523
29.4.6.1 PPCID[0:3]—PF PCI Express Capability ID Register....1523
29.4.6.2 PPCP[0:3]—PF PCI Express Next Capability Pointer Register....1524
29.4.6.3 PPCR[0:3]—PF PCI Express Capabilities Register....1524
29.4.6.4 PPDCAP[0:3]—PF PCI Express Device Capabilities Register....1525
29.4.6.5 PPDCNTL[0:3]—PF PCI Express Device Control Register....1527

29.4.6.6 PPDSTAT[0:3]—PF PCI Express Device Status Register 1528
29.4.6.7 PLCAPR[0:3]—PF Link Capabilities Register 1529
29.4.6.8 PLCNTLR[0:3]—PF Link Control Register 1531
29.4.6.9 PLSR[0:3]—PF Link Status Register 1532
29.4.6.10 PDCAPR2[0:3]—PF Device Capabilities 2 Register 1534
29.4.6.11 PDCNTR2[0:3]—PF Device Control 2 Register 1535
29.4.6.12 PLCNTLR2[0:3]—PF Link Control 2 Register 1536
29.4.6.13 PLSR2[0:3]—PF Link Status 2 Register 1538
29.4.7 Vital Product Data (VPD) Registers 1538
29.4.7.1 VPDCID[0:3]—VPD Capability ID Register 1539
29.4.7.2 VPDNCP[0:3]—VPD Next Capability Pointer Register 1539
29.4.7.3 VPDADDR[0:3]—VPD Address Register 1540
29.4.7.4 VPDDATA[0:3]—VPD Data 1540
29.5 PCIe* Extended Configuration Space 1541
29.5.1 Advanced Error Reporting (AER) Capability 1541
29.5.2 PF Advanced Error Reporting Capability Structure 1542
29.5.2.1 PPCIEAERCAPID[0:3]—PF PCI Express AER Capability ID Register 1542
29.5.2.2 PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register 1542
29.5.2.3 PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register 1543
29.5.2.4 PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register 1544
29.5.2.5 PPAERCS[0:3]—PF PCI Express AER Correctable Error Register 1545
29.5.2.6 PPAERCM[0:3]—PF PCI Express AER Correctable Error Mask Register ... 1546
29.5.2.7 PPAERCTLCAP[0:3]—PF PCI Express AER Control and Capability Register 1546
29.5.2.8 PPAERHDRLOG0[0:3]—PF PCI Express AER Header Log 0 Register 1547
29.5.2.9 PPAERHDRLOG1[0:3]—PF PCI Express AER Header Log 1 Register 1547
29.5.2.10 PPAERHDRLOG2[0:3]—PF PCI Express AER Header Log 2 Register 1548
29.5.2.11 PPAERHDRLOG3[0:3]—PF PCI Express AER Header Log 3 Register 1548
29.5.3 PF Alternative Routing-ID Extended Capability Structure 1549
29.5.3.1 PARIDHDR[0:3]—PF Alternative Routing ID Capability Header 1549
29.5.3.2 PFARICAP0—PF ARI Capabilities Register 1550
29.5.3.3 PFARICAP1—PF ARI Capabilities Register 1551
29.5.3.4 PFARICAP2—PF ARI Capabilities Register 1552
29.5.3.5 PFARICAP3—PF ARI Capabilities Register 1552
29.5.3.6 PARIDCTL[0:3]—PF Alternative Routing ID Control Register 1553
29.5.4 TLP Processing Hint Requester (TPH) Capability 1553
29.5.4.1 TPH CAP ID (0x1A0; RO) 1553
29.5.4.2 TPH Requester Capabilities (0x1A4; RO) 1553
29.5.4.3 TPH Requester Control (0x1A8; R/W) 1553
29.5.4.4 TPH Steering table (0x1AC - 0x1B8; R/W) 1553
29.5.4.5 Latency Tolerance Requirement Reporting (LTR) Capability 1553
29.5.4.6 LTR CAP ID (0x1C0; RO) 1553
29.5.4.7 LTR Capabilities (0x1C4; RW) 1554
29.5.5 I/O Space 1554

Test Features - Volume 3 of 4 1555

30.0 PCH Global Test Features 1557
30.1 JTAG 1557
30.1.1 JTAG Functions Overview 1557
30.1.2 PCH TAP Interfaces 1557
30.1.3 TAP Controller Operation and State Diagram 1558

Technical Specifications - Volume 4 of 4 1560

31.0 System Clocks 1562

31.1 External Clock Requirements 1562

32.0 Signal Descriptions 1563
32.1 Name Convention 1563
32.2 Intel® Communications Chipset 89xx Series SKU Naming 1564
32.3 JTAG Boundary Scan Chain (BSC) 1149.1 and 1149.6 Chain 1566
32.4 Direct Media Interface 1566
32.5 PCI Express* EndPoint 1567
32.6 Gigabit Ethernet Interface 1570
32.7 EndPoint Management SMBus Interface (Slave) 1585
32.8 PCI Express* Root Complex 1586
32.9 Serial ATA Interface 1587
32.10 LPC Interface 1589
32.11 USB* Interface 1590
32.12 UART Interface 1591
32.13 Host SMBus (Master) Interface 1594
32.14 Serial Peripheral Interface Boot Interface 1596
32.15 Interrupt Interface 1596
32.16 Processor Interface 1597
32.17 Power Management Interface 1598
32.18 Thermal Sensor Current Reference 1601
32.19 Miscellaneous Interface 1601
32.20 General Purpose Input/Output Interface 1602
32.21 Real Time Clock (RTC) Interface 1611
32.22 System Input Clock 1613
32.23 JTAG Interface 1615
32.24 Strapping Signals 1617
32.25 Reserved Signals 1624
32.26 Power and Ground Signals 1629

33.0 Electrical Characteristics 1634
33.1 Absolute Maximum and Minimum Ratings 1634
33.2 Recommended Power Supply Range 1635
33.3 Maximum ICC Supply Current 1636
33.4 Power Supply Pin Groupings 1637
33.5 General DC Characteristics 1640
33.6 PCI Express* Root Complex and DMI Gen1 DC/AC Characteristics 1644
33.7 PCIe* Gen2 EndPoint DC/AC Specification 1647
33.7.1 PCIe* Specification - Input Clock 1649
33.8 SATA Gen1/Gen2 DC/AC Characteristics 1650
33.9 Gigabit Ethernet SGMII DC/AC Characteristics 1651
33.10 SerDes DC/AC Specification 1652
33.11 CRU Clock DC/AC Specification 1653
33.12 SATA, DMI, and PCIe* Clocks DC/AC Specification 1655
33.13 AC Characteristics 1659
33.13.1 I2C/SFP AC Specification 1666
33.13.2 MDIO DC Specification 1667
33.13.3 MDIO AC Specification 1668
33.13.4 EEPROM AC Specification 1669
33.13.5 UART AC Specification 1671
33.13.6 JTAG AC Specification 1671
33.14 AC Timing Diagrams 1672

34.0 Thermal Specifications and Design Considerations 1678

35.0 Packaging Information 1679
35.1 Package Introduction 1679

35.2 Ball Map Information 1679
35.2.1 Ball Map Pin Lists 1680
35.2.2 Ball Map Illustrations 1696
35.3 Package Mechanical Information 1710

Figures

2-1 PCH Modes 98
2-2 PCH Block Diagram 99
3-1 Device-Centric Logical View of PCH Devices 102
3-2 Attaching PCH to the PCI Fabric (Logical Perspective) 107
4-1 LPC Interface Diagram 116
4-2 DMA Controller 120
4-3 DMA Request Assertion through LDRQ# 123
4-4 System Management Bus (SMBus) Interface 161
4-5 SMBus Block-Size Configuration Register Read 180
4-6 SMBus Block-Size Memory Register Read 181
4-7 SMBus Word-Size Configuration Register Read 181
4-8 SMBus Word-Size Memory Register Read 182
4-9 SMBus Byte-Size Configuration Register Read 182
4-10 SMBus Byte-Size Memory Register Read 183
4-11 SMBus Block-Size Configuration Register Write 184
4-12 SMBus Block-Size Memory Register Write 184
4-13 SMBus Word-Size Configuration Register Write 184
4-14 SMBus Word-Size Memory Register Write 184
4-15 SMBus Configuration (Byte Write, PEC Enabled) 185
4-16 SMBus Memory (Byte Write, PEC Enabled) 185
4-17 Example UART Data Frame 190
4-18 WDT Block Diagram 193
4-19 Serial Post over GPIO Reference Circuit 196
4-20 1-byte Serial Write with a Data Byte of 5Ah 198
4-21 Serial Data Transmitted Over the SGPIO Interface 206
4-22 EHCI with USB* 2.0 with Rate Matching Hub 219
4-23 MSI Packet Header 222
4-24 Flash Partition Boundary 224
4-25 Flash Descriptor Sections 225
19-1 PCIe* EP Interface Block Diagram 875
19-2 EP Functional Description Block Diagram 876
19-3 EP Block Diagram 890
19-4 Ring Block Diagram 891
21-1 High-Level View of GbE Interface Connectivity in PCH Implementation 994
22-7 SMBus ARP Flow 1014
22-9 GbE Controller Loopback Modes 1038
23-1 Power-Up - General Flow 1040
23-2 Power-Up Timing Diagram 1041
25-1 Power Management State Diagram 1096
25-2 Power Up (Off to Dup to D0u to D0a) 1101
25-3 Transition From D0a to D3 and Back Without PCIE_EP_RST# 1102
25-4 Transition From D0a to D3 and Back With PCIE_EP_RST# 1103
25-5 Transition From D0a to Dr and Back Without Transition to D3 1104
26-1 Stages in Packet Filtering 1113
26-2 Receive Queuing Flow (Virtualization) 1116
26-3 RSS Block Diagram 1120
26-12 Receive Filtering Flow Chart 1126

26-13 Host MAC Address Receive Filtering Flow Chart 1127
26-14 VLAN Filtering 1129
26-15 Platform Manageability Filtering 1130
26-16 Receive Descriptor Ring Structure 1141
26-17 Header Splitting 1143
26-18 Header Replication 1144
26-19 Transmit Descriptor Ring Structure 1161
26-20 Cause Mapping in Legacy Mode 1178
26-21 Cause Mapping in MSI-X Mode 1179
26-22 Interrupt Throttle Flow Diagram 1185
26-23 Case A: Heavy Load, Interrupts Moderated 1186
26-24 Light Load, Interrupts Immediately on Packet Receive 1186
26-25 Packet Reception Decision Table 1191
26-26 Diagram of DCA Implementation on FSB System 1194
26-27 PCIe* Message Format for DCA 1195
26-28 Pool List Selection - Replication Enabled 1200
26-29 Pool List Selection - Replication Disabled 1201
26-32 Sync Flow and Offset Calculation 1208
26-33 Time Stamp Point 1211
26-34 1220
27-1 Controller to BMC Connectivity Through a SMBus-only Connection 1221
27-2 Manageability Decision Filters 1244
28-1 Multicast Table Array 1342
29-1 GbE Controller PCI Configuration Registers 1490
30-1 TAP Connectivity 1558
30-2 TAP Controller State Diagram 1559
32-1 Interface Signals Block Diagram1 1565
33-1 PCI Express Gen1 Transmitter Eye 1645
33-2 PCI Express Gen1 Receiver Eye 1646
33-3 PCI Express Transmitter Eye 1648
33-4 PCI Express Receiver Eye 1649
33-5 CRU Differential Clock Waveform 1654
33-6 CRU Differential Clock Cross-Point Specification 1654
33-7 SATA, DMI, and PCIe* Clocks Differential Clock Waveform 1656
33-8 Single-Ended Measurement Point for Absolute Cross Point and Swing 1657
33-9 Single-Ended Clock Measurement Points for Delta Cross Point 1657
33-10 Differential Clock Cross Point Specification 1657
33-11 Differential Measurement Point for Duty Cycle and Period 1658
33-12 Differential Measurement Point for Rise and Fall Time 1658
33-13 Differential Measurement Point for Ringback 1658
33-14 Digital I/O Output Timing Diagram 1659
33-15 Digital I/O Input Timing Diagram 1660
33-16 I^2C I/F Timing Diagram 1666
33-17 MDIO Input AC Timing Diagram 1668
33-18 MDIO Output AC Timing Diagram 1669
33-19 EEPROM Timing Diagram 1670
33-20 JTAG AC Timing Diagram 1672
33-21 Clock Cycle Time 1672
33-22 Clock Timing 1672
33-24 Setup and Hold Times 1673
33-25 Float Delay 1673
33-26 Pulse Width 1673
33-23 Valid Delay from Rising Clock Edge 1673
33-27 Output Enable Delay 1674
33-28 USB Rise and Fall Times 1674

33-29 USB Jitter 1674
33-30 USB EOP Width 1675
33-31 SMBus Transaction 1675
33-32 SMBus Timeout 1675
33-33 SPI Timings 1676
33-34 Measurement Points for Differential Waveforms 1677
35-1 Mechanical Package 1710

Tables

1-1 Intel® Communications Chipset 89xx Series SKUs 85
1-2 Summary of Differences between DH89xxCC and DH89xxCL SKUs 86
1-3 Referenced Documents 88
1-4 Related Websites 88
1-5 Acronym Table 89
1-6 Glossary Table 92
2-1 PCH External Interface Summary 100
3-1 Supported Operations by Memory Type 105
3-2 Address Space Sizes of PCIe* Endpoint-Attached Devices 106
3-3 PCH PCI Device Summary 108
3-4 EP PCI Device Summary 108
3-5 DH89xxCC GbE PCI Device ID Summary 109
3-6 PCI Configuration Header Support for Type 0 Headers in PCIe* Endpoint Devices 110
4-1 MSI vs. PCI IRQ Actions 112
4-2 LPC Cycle Types Supported 117
4-3 Start Field Bit Definitions 117
4-4 Cycle Type Bit Definitions 118
4-5 Transfer Size Bit Definition 118
4-6 SYNC Bit Definition 118
4-7 DMA Transfer Size 122
4-8 Address Shifting in 16-Bit I/O DMA Transfers 122
4-9 Counter Operating Modes 127
4-10 Interrupt Controller Core Connections 129
4-11 Interrupt Status Registers 130
4-12 Content of Interrupt Vector Byte 130
4-13 APIC Interrupt Mapping1 136
4-14 Stop Frame Explanation 138
4-15 Data Frame Format 139
4-16 Configuration Bits Reset by RTCRST# Assertion 141
4-17 INIT# Going Active 143
4-18 NMI Sources 144
4-19 General Power States for Platform Systems 145
4-20 State Transition Rules 146
4-21 System Power Plane 146
4-22 Causes of SMI and SCI 147
4-23 Sleep Types 150
4-24 Causes of Wake Events 151
4-25 GPI Wake Events 151
4-26 Transitions Due to Power Failure 152
4-27 Transitions Due to Power Button 153
4-28 Transitions Due to RI# Signal 154
4-29 Write Only Registers with Read Paths in ALT Access Mode 156
4-30 PIC Reserved Bits Return Values 157
4-31 Register Write Accesses in ALT Access Mode 158
4-32 Causes of Host and Global Resets 160
4-33 I^2C Block Read 165
4-34 Enable for SMBALERT# 167
4-35 Enables for SMBus Slave Write and SMBus Host Events 167

4-36 Enables for the Host Notify Command 168
4-37 Slave Write Registers 169
4-38 Command Types 170
4-39 Slave Read Cycle Format 170
4-40 Data Values for Slave Read Registers 171
4-41 Host Notify Format 173
4-42 Event Transitions that Cause Messages 175
4-43 SMBus Command Encoding 176
4-44 Internal SMBus Protocol Stack 177
4-45 SMBus Slave Address Format 178
4-46 Memory Region Address Field 178
4-47 Status Field Encoding for SMBus Reads 179
4-48 Address Map 187
4-49 Supported LPC Cycle Types 187
4-50 I/O Sync Bits Description 188
4-51 UART Clock Divider Support 189
4-52 Baud Rate Example 189
4-53 UART Register/Signal Reset States 190
4-54 Multi-Activity LED Message Type 205
4-55 Legacy Replacement Routing 207
4-56 USB* EHCI Features 210
4-57 Debug Port Behavior 214
4-58 Region Size Versus Erase Granularity of Flash Components 224
4-59 Region Access Control Table 226
4-60 Hardware Sequencing Commands and Opcode Requirements 228
4-61 Flash Protection Mechanism Summary 229
4-62 Recommended Pinout for 8-Pin Serial Flash Device 230
4-63 Recommended Pinout for 16-Pin Serial Flash Device 231
5-1 233
5-2 Fixed I/O Ranges Decoded 233
5-3 Variable I/O Decode Ranges 235
5-4 Memory Decode Ranges (From CPU Perspective) 236
6-1 PCI Configuration Registers (LPC I/F—B0:D31:F0) 239
6-2 Offset 00h: ID: Vendor Identification Register 241
6-3 Offset 02h: DID—Device Identification Register (LPC I/F—B0:D31:F0) 241
6-4 Offset 04h: ID—PCICMD—PCI COMMAND Register (LPC I/F—B0:D31:F0) 242
6-5 Offset 06h: ID—PCISTS—PCI Status Register (LPC I/F—B0:D31:F0) 243
6-6 Offset 08h: RID—Revision Identification Register (PCI-PCI—B0:D31:F0) 244
6-7 Offset 09h: PI—Programming Interface Register (PCI-PCI—B0:D31:F0) 244
6-8 Offset 0Ah: SCC—Sub Class Code Register (PCI-PCI—B0:D31:F0) 244
6-9 Offset 0Bh: BCC—Base Class Code Register (LPC I/F—B0:D31:F0) 245
6-10 Offset 0Dh: PLT—Primary Latency Timer Register (LPC I/F—B0:D31:F0) 245
6-11 Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—B0:D31:F0) 245
6-12 Offset 2Ch: SS—Sub System Identifiers Register (LPC I/F—B0:D31:F0) 246
6-13 Offset 40h: PMBASE—ACPI Base Address Register (LPC I/F—B0:D31:F0) 246
6-14 Offset 44h: ACPI_CNTL—ACPI Control Register (LPC I/F—B0:D31:F0) 247
6-15 Offset 48h: GPIOBASE—GPIO Base Address Register (LPC I/F—B0:D31:F0) 248
6-16 Offset 4Ch: GC—GPIO Control Register (LPC I/F—B0:D31:F0) 249
6-17 Offset 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control Register (LPC I/F—B0:D31:F0) 250
6-18 Offset 64h: SIRQ_CNTL—Serial IRQ Control Register (LPC I/F—B0:D31:F0) 251
6-19 Offset 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control Register (LPC I/F—B0:D31:F0) 252
6-20 Offset 6Ch: LPC_IBDF—IOxAPIC Bus:Device:Function (LPC I/F—B0:D31:F0) 253
6-21 Offset 70h: LPC_HnBDF—HPET n Bus:Device:Function (LPC I/F—B0:D31:F0) 254
6-22 Offset 80h: LPC_I/O_DEC—I/O Decode Ranges Register (LPC I/F—B0:D31:F0) 255
6-23 Offset 82h: LPC_EN—LPC I/F Enables Register (LPC I/F—B0:D31:F0) 256
6-24 Offset 84h: GEN1_DEC—LPC I/F Generic Decode Range 1 Register (LPC I/F—B0:D31:F0) 258

6-25 Offset 88h: GEN2_DEC—LPC I/F Generic Decode Range 2 Register (LPC I/F—B0:D31:F0) 259
6-26 Offset 8Ch: GEN3_DEC—LPC I/F Generic Decode Range 3 Register (LPC I/F—B0:D31:F0) 260
6-27 Offset 90h: GEN4_DEC—LPC I/F Generic Decode Range 4 Register (LPC I/F—B0:D31:F0) 261
6-28 Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control (LPC I/F—B0:D31:F0) 262
6-29 Offset 98h: LGMR—LPC I/F Generic Memory Range (LPC I/F—B0:D31:F0) 264
6-30 Offset DCh: BIOS_CNTL—BIOS Control Register (LPC I/F—B0:D31:F0) 265
6-31 Offset E0h: FDCAP—Feature Detection Capability ID (LPC I/F—B0:D31:F0) 266
6-32 Offset E2h: FDLEN—Feature Detection Capability Length (LPC I/F—B0:D31:F0) 266
6-33 Offset E3h: FDVER—Feature Detection Version (LPC I/F—B0:D31:F0) 267
6-34 Offset E4h: FDVCT—Feature Vector (LPC I/F—B0:D31:F0) 267
6-35 Offset F0h: RCBA—Root Complex Base Address Register (LPC I/F—B0:D31:F0) 268
6-36 DMA I/O Registers (LPC I/F—B0:D31:F0) 268
6-37 Offset 00h: DMABASE_CA—DMA Base and Current Address Registers (LPC I/F — B0:D31:F0) 269
6-38 Offset 01h: DMABASE_CC—DMA Base and Current Count Registers (LPC I/F—B0:D31:F0) 270
6-39 Offset 87h: DMAMEM_LP—DMA Memory Low Page Registers (LPC I/F—B0:D31:F0) ... 271
6-40 Offset 08h: DMACMD—DMA Command Register (LPC I/F—B0:D31:F0) 272
6-41 Offset 08h: DMASTA—DMA Status Register (LPC I/F—B0:D31:F0) 273
6-42 Offset 0Ah: DMA_WRSMSK—DMA Write Single Mask Register (LPC I/F—B0:D31:F0) .. 274
6-43 Offset 08h: DMACH_MODE—DMA Channel Mode Register (LPC I/F—B0:D31:F0) 275
6-44 Offset 0Ch: DMA Clear Byte Pointer Register (LPC I/F—B0:D31:F0) 276
6-45 Offset 0Dh: DMA Master Clear Register (LPC I/F—B0:D31:F0) 276
6-46 Offset 0Eh: DMA Master Clear Register (LPC I/F—B0:D31:F0) 276
6-47 Offset 0Fh: DMA_WRMSK—DMA Write All Mask Register (LPC I/F—B0:D31:F0) 277
6-48 Timer I/O Registers 277
6-49 Offset 43h: TCW—Timer Control Word Register (LPC I/F—B0:D31:F0) 278
6-50 Offset 40h: SBYTE_FMT—Interval Timer Status Byte Format Register (LPC I/F—B0:D31:F0) 280
6-51 Offset 40h: Counter Access Ports Register (LPC I/F—B0:D31:F0) 281
6-52 PIC Registers 282
6-53 Offset 20h: Master PIC ICW1—Master Initialization Command Word 1 Register 283
6-54 Offset 21h: Master PIC ICW2—Master Initialization Command Word 2 Register 284
6-55 Offset 21h: Master PIC ICW3—Master Initialization Command Word 3 Register 285
6-56 Offset A1h: Slave PIC ICW3—Slave Initialization Command Word 3 Register..... 285
6-57 Offset 021h: Master PIC ICW4—Master Initialization Command Word 4 Register..... 286
6-58 Offset 021h: Master PIC OCW1—Master Operational Control Word 1 (Interrupt Mask) Register 287
6-59 Offset 020h: Master PIC OCW2—Master Operational Control Word 2 Register..... 288
6-60 Offset 020h: Master PIC OCW3—Master Operational Control Word 3 Register..... 289
6-61 Offset 4D0h: Master PIC ELCR1—Master Controller Edge/Level Triggered Register..... 290
6-62 Offset 4D1h: Slave PIC ELCR2—Slave Controller Edge/Level Triggered Register 291
6-63 APIC Direct Registers 292
6-64 IND—Index Register 292
6-65 DAT—Data Register 293
6-66 EOIR—EOI Register..... 294
6-67 APIC Indirect Registers..... 294
6-68 Offset 00h: ID—Identification Register..... 295
6-69 Offset 01h: VER—Version Register 296
6-70 Offset 10h: REDIR_TBL0—Redirection Table 0 297
6-71 RTC I/O Registers..... 299
6-72 RTC (Standard) RAM Bank 300
6-73 Real Time Clock Registers..... 300
6-74 Offset 0Ah: RTC_REGA—Register A 301
6-75 Offset 0Bh: RTC_REGB—Register B (General Configuration) 302
6-76 Offset OCh: RTC_REGC—Register C (Flag Register) 303

6-77 Offset ODh: D—Register D (Flag Register) 304
6-78 Processor Interface PCI Register Address Map (LPC I/F—B0:D31:F0)..... 305
6-79 Offset 61h: NMI_SC—NMI Status and Control Register (LPC I/F—B0:D31:F0)..... 306
6-80 Offset 70h: NMI_EN—NMI Enable (and Real Time Clock Index) Register (LPC I/F—B0:D31:F0) 307
6-81 Offset 92h: PORT92—Fast A20 and Init Register (LPC I/F—B0:D31:F0) 307
6-82 Offset F0h: COPROC_ERR—Coprocessor Error Register (LPC I/F—B0:D31:F0)..... 308
6-83 Offset CF9h: RST_CNT—Reset Control Register (LPC I/F—B0:D31:F0) 308
6-84 Power Management PCI Register Address Map (PM—B0:D31:F0) 310
6-85 Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0) . 311
6-86 Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0) 313
6-87 Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0) . 315
6-88 Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register 318
6-89 Offset A9h: Chipset Initialization Register 4 (PM—B0:D31:F0) 318
6-90 Offset ABh: BM_BREAK_EN Register (PM—B0:D31:F0) 319
6-91 Offset ACh: PMIR—Power Management Initialization Register (PM—B0:D31:F0) 319
6-92 Offset B8h: GPIO_ROUT—GPIO Routing Control Register (PM—B0:D31:F0)..... 320
6-93 APM Register Map 320
6-94 Offset B2h: APM_CNT—Advanced Power Management Control Port Register 321
6-95 Offset B3h: APM_STS—Advanced Power Management Status Port Register 321
6-96 ACPI and Legacy I/O Register Map..... 322
6-97 Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register 323
6-98 Offset PMBASE + 02h: PM1_EN—Power Management 1 Enable Register..... 326
6-99 Offset PMBASE + 04h: PM1_CNT—Power Management 1 Control 327
6-100 Offset PMBASE + 08h: PM1_TMR—Power Management 1 Timer Register..... 328
6-101 Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register 329
6-102 Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register 331
6-103 Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register..... 333
6-104 Offset PMBASE + 34h: SMI_STS—SMI Status Register 335
6-105 Offset PMBASE +38h: ALT_GP_SMI_EN—Alternate GPI SMI Enable Register..... 338
6-106 Offset PMBASE +3Ah: ALT_GP_SMI_STS—Alternate GPI SMI Status Register 338
6-107 Offset PMBASE +3Ch: UPRWC—USB Per-Port Registers Write Control..... 339
6-108 Offset PMBASE +42h: GPE_CNTL—General Purpose Control Register..... 340
6-109 Offset PMBASE +44h: DEVACT_STS—Device Activity Status Register 341
6-110 Offset PMBASE +50h: PM2_CNT—Power Management 2 Control 342
6-111 TCO I/O Register Address Map 342
6-112 Offset TCOBASE +00h: TCO_RLD—TCO Timer Reload and Current Value Register 343
6-113 Offset TCOBASE +02h: TCO_DAT_IN—TCO Data In Register..... 343
6-114 Offset TCOBASE +03h: TCO_DAT_OUT—TCO Data Out Register 344
6-115 Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register 344
6-116 Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register 346
6-117 Offset TCOBASE +08h: TCO1_CNT—TCO1 Control Register..... 348
6-118 Offset TCOBASE +0Ah: TCO2_CNT—TCO2 Control Register..... 349
6-119 Offset TCOBASE +0Ch and Offset TCOBASE +0Dh: TCO_MESSAGE1 and TCO_MESSAGE2 Registers 350
6-120 Offset TCOBASE + 0Eh: TCO_WDCNT—TCO Watchdog Control Register..... 350
6-121 Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register 351
6-122 Offset TCOBASE + 12h:TCO_TMR—TCO Timer Initial Value Register..... 351
6-123 General Purpose I/O Signals 352
6-124 Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register..... 353
6-125 Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register 354
6-126 Offset GPIOBASE + 0Ch:GP_LVL—GPIO Level for Input or Output Register 354
6-127 Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register..... 355
6-128 Offset GPIOBASE + 1Ch: GP_SER_BLINK—GP Serial Blink Data..... 356
6-129 Offset GPIOBASE + 20h: GP_SB_CMDSTS—GP Serial Blink Command Status 357
6-130 Offset GPIOBASE + 24h: GP_SB_DATA—GP Serial Blink Data..... 358

6-131 Offset GPIOBASE + 28h: GPI_NMI_EN—GPI NMI Enable 358
6-132 Offset GPIOBASE + 2Ah: GPI_NMI_STS—GPI NMI Status 359
6-133 Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register 359
6-134 Offset GPIOBASE + 30h: GPIO_USE_SEL2—GPIO Use Select 2 Register 360
6-135 Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register 361
6-136 Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register 361
6-137 Offset GPIOBASE + 40h: GPIO_USE_SEL3—GPIO Use Select 3 Register 362
6-138 Offset GPIOBASE + 44h: GP_IO_SEL3—GPIO Input/Output Select 3 Register 363
6-139 Offset GPIOBASE + 48h: GP_LVL3—GPIO Level for Input or Output 3 Register 364
6-140 Offset GPIOBASE + 60h: GP_RST_SEL1—GPIO Reset Select 365
6-141 Offset GPIOBASE + 64h: GP_RST_SEL2—GPIO Reset Select 366
6-142 Offset GPIOBASE + 68h: GP_RST_SEL3—GPIO Reset Select 366
7-1 Chipset Configuration Registers 367
7-2 Offset 0014h: V0CTL—Virtual Channel 0 Resource Control Register 370
7-3 Offset 001Ah: V0STS—Virtual Channel 0 Resource Status Register 371
7-4 Offset 001Ch: V1CAP—Virtual Channel 1 Resource Capability Register 371
7-5 Offset 0020h: V1CTL—Virtual Channel 1 Resource Control Register 372
7-6 Offset 0026h: V1STS—Virtual Channel 1 Resource Status Register 372
7-7 Offset 0050h: CIR0—Chipset Initialization Register 0 373
7-8 Offset 0088h: CIR1—Chipset Initialization Register 1 373
7-9 Offset 00ACh: REC—Root Error Command Register 374
7-10 Offset 01A0h: ILCL—Internal Link Capabilities List Register 374
7-11 Offset 01A4h: LCAP—Link Capabilities Register 375
7-12 Offset 01A8h: LCTL—Link Control Register 375
7-13 Offset 01AAh: LSTS—Link Status Register 376
7-14 Offset 0220h: BCR—Backbone Configuration Register 376
7-15 Offset 0224h: RPC—Root Port Configuration Register 377
7-16 Offset 0234h: DMIC—DMI Control Register 378
7-17 Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports ... 378
7-18 Offset 0290h: Reserved 379
7-19 Offset 1D40H: CIR5—Chipset Initialization Register 5 380
7-20 Offset 1E00h: TRSR—Trap Status Register 380
7-21 Offset 1E10h: TRCR—Trapped Cycle Register 381
7-22 Offset 1E18h: TWDR—Trapped Write Data Register 381
7-23 Offset 1E80h: IOTRn—I/O Trap Register (0-3) 382
7-24 Offset 2010h: DMC—DMI Miscellaneous Control Register 383
7-25 Offset 2024h: CIR6—Chipset Initialization Register 6 383
7-26 Offset 2324h: DMC2—DMI Miscellaneous Control Register 2 384
7-27 Offset 60h: SBI Unified AFE Address Register (SATA–B0:D31:F2) 384
7-28 Offset 64h: SBI Unified AFE Data Register (SATA–B0:D31:F2) 385
7-29 Offset 68h: SBI Unified AFE Status Register (SATA–B0:D31:F2) 386
7-30 Offset 3000h: TCTL—TCO Configuration Register 387
7-31 Offset 3100h: D31IP—Device 31 Interrupt Pin Register 388
7-32 Offset 3108h: D29IP—Device 29 Interrupt Pin Register 389
7-33 Offset 310Ch: D28IP—Device 28 Interrupt Pin Register 390
7-34 Offset 3124h: D22IP—Device 22 Interrupt Pin Register 391
7-35 Offset 3140h: D31IR—Device 31 Interrupt Route Register 392
7-36 Offset 3144h: D29IR—Device 29 Interrupt Route Register 393
7-37 Offset 3146h: D28IR—Device 28 Interrupt Route Register 394
7-38 Offset 315Ch: D22IR—Device 22 Interrupt Route Register 395
7-39 Offset 31FEh: OIC—Other Interrupt Control Register 396
7-40 Offset 3310h: PRSTS—Power and Reset Status 397
7-41 Offset 3314h: CIR7—Chipset Initialization Register 7 398
7-42 Offset 3324h: CIR8—Chipset Initialization Register 8 398
7-43 Offset 3330h: CIR9—Chipset Initialization Register 9 398
7-44 Offset 3340h: CIR10—Chipset Initialization Register 10 399
7-45 Offset 3350h: CIR13—Chipset Initialization Register 13 399
7-46 Offset 3368h: CIR14—Chipset Initialization Register 14 399

7-47 Offset 3378h: CIR15—Chipset Initialization Register 15 400
7-48 Offset 3388h: CIR16—Chipset Initialization Register 16 400
7-49 Offset 33A0h: CIR17—Chipset Initialization Register 17 400
7-50 Offset 33A8h: CIR18—Chipset Initialization Register 18 401
7-51 Offset 33C0h: CIR19—Chipset Initialization Register 19 401
7-52 Offset 33CCh: CIR20—Chipset Initialization Register 20 401
7-53 Offset 33D0h: CIR21—Chipset Initialization Register 21 402
7-54 Offset 33D4h: CIR22—Chipset Initialization Register 22 402
7-55 Offset 3400h: RC—RTC Configuration Register 403
7-56 Offset 3404h: HPTC—High Precision Timer Configuration Register 404
7-57 Offset 3410h: GCS—General Control and Status Register 405
7-58 Offset 3414h: BUC—Backed Up Control Register 407
7-59 Offset 3418h: FD—Function Disable Register 408
7-60 Offset 341Ch: CG—Clock Gating 410
7-61 Offset 3420h: FDSW—Function Disable SUS Well 411
7-62 Offset 3428h: FD2—Function Disable 2 411
7-63 Offset 3500h: USBIR[0:5]—USB Initialization Register [0-5] 412
7-64 Offset 3564h: USBIRC—USB Initialization Register C 413
7-65 Offset 3570h: USBIRA—USB Initialization Register A 413
7-66 Offset 357Ch: USBIRB—USB Initialization Register B 414
7-67 Offset 3590h: MISCCTL—Miscellaneous Control Register 415
7-68 Offset 359Ch: PDO—USB Port Disable Override 416
7-69 Offset 35A0h: USBOCM1—Overcurrent MAP Register 1 417
7-70 Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register 418
8-1 SATA* Controller PCI Register Address Map 420
8-2 Offset 00h: Vendor Identification Register (SATA–B0:D31:F2) 422
8-3 Offset 02h: Device Identification Register (SATA–B0:D31:F2) 422
8-4 Offset 04h: PCI Command Register (SATA–B0:D31:F2) 423
8-5 Offset 06h: PCI Status Register (SATA–B0:D31:F2) 424
8-6 Offset 08h: Revision Identification Register (SATA–B0:D31:F2) 425
8-7 Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 01h 425
8-8 Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 06h 426
8-9 Offset 0Ah: Sub Class Code Register (SATA–B0:D31:F2) 427
8-10 Offset 0Bh: Base Class Code Register (B0:D31:F2) 427
8-11 Offset 0Dh: Primary Master Latency Timer Register (B0:D31:F2) 427
8-12 Offset 0Eh: Header Type (SATA–B0:D31:F2) 428
8-13 Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F2) 428
8-14 Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F2) 429
8-15 Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1) 429
8-16 Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F2) 430
8-17 Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F2) 430
8-18 Offset 24h: When SCC is Not 01h (SATA–B0:D31:F2) 431
8-19 Offset 24h: When SCC is 01h (SATA–B0:D31:F2) 432
8-20 Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F2) 432
8-21 Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F2) 433
8-22 Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F2) 433
8-23 Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F2) 433
8-24 Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F2) 434
8-25 Offset 40h: IDE Timing Register (SATA–B0:D31:F2) 434
8-26 Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F2) 435
8-27 Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F2) 435
8-28 Offset 4Ah: IDE I/O Configuration Register (SATA–B0:D31:F2) 436
8-29 Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F2) 436
8-30 Offset 72h: PCI Power Management Capabilities Register (SATA–B0:D31:F2) 437
8-31 Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F2) . 438
8-32 Offset 80h: Message Signaled Interrupt Capability Identification (SATA–B0:D31:F2) . 439
8-33 Offset 82h: Message Signaled Interrupt Message Control (SATA–B0:D31:F2) 440
8-34 Offset 84h: Message Signaled Interrupt Message Address (SATA–B0:D31:F2) 441

8-35 Offset 88h: Message Signaled Interrupt Message Data (SATA–B0:D31:F2) 441
8-36 Offset 90h: MAP—Address Map Register (SATA–B0:D31:F2) 442
8-37 Offset 92h: PCS—Port Control and Status Register (SATA–B0:D31:F2) 444
8-38 Offset 94h: SCLKCG—SATA Clock Gating Control Register (SATA–B0:D31:F2) 445
8-39 Offset 9Ch: SCLKGC—SATA Clock General Configuration Register (SATA–B0:D31:F2) 446
8-40 Offset A0h: SIRI—SATA Indexed Registers Index (SATA–B0:D31:F2) 447
8-41 Offset A4h: STRD—SATA Indexed Register Data (SATA–B0:D31:F2) 447
8-42 Offset A8h: SATACR0—SATA Capability Register 0 (SATA–B0:D31:F2) 448
8-43 Offset ACh: SATACR1—SATA Capability Register 1 (SATA–B0:D31:F2) 449
8-44 Offset B0h: FLRCID—FLR Capability ID (SATA–B0:D31:F2) 450
8-45 Offset B2h: FRLCLV—FLR Capability Length and Version (SATA–B0:D31:F2) 450
8-46 Offset B2h: FLRCLV—FLR Capability Length and Version (SATA–B0:D31:F2) 451
8-47 Offset B4h: FLRC—FLR Control (SATA–B0:D31:F2) 451
8-48 Offset C0h: ATC—APM Trapping Control Register (SATA–B0:D31:F2) 452
8-49 Offset C4h: ATS—APM Trapping Status Register (SATA–B0:D31:F2) 452
8-50 Offset D0h: SP—Scratch Pad Register (SATA–B0:D31:F2) 453
8-51 Offset E0h: BFCS—BIST FIS Control/Status Register (SATA–B0:D31:F2) 453
8-52 Offset E4h: FBTD1—BIST FIS Transmit Data1 Register (SATA–B0:D31:F2) 455
8-53 Offset E8h: BFTD2—BIST FIS Transmit Data2 Register (SATA–B0:D31:F2) 455
8-54 SATA* Indexed Registers 456
8-55 Offset 18h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 457
8-56 Offset 1Ch: SATA* Indexed Registers Index (SATA* Test Mode Enable Register) (SATA–B0:D31:F2) 457
8-57 Offset 28h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 458
8-58 Offset 3Eh: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 458
8-59 Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 459
8-60 Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 459
8-61 Offset 68h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 460
8-62 Offset 78h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 460
8-63 Offset 84h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 461
8-64 Offset 88h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 461
8-65 Offset 8Ch: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 462
8-66 Offset 94h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 462
8-67 Offset A0h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 463
8-68 Offset A8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 463
8-69 Offset C4h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 464
8-70 Offset C8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2) 464
8-71 Bus Master IDE I/O Register Address Map 465
8-72 Offset 00h: Bus Master IDE Command Register Primary (B0:D31:F2) 466
8-73 Offset 02h: Bus Master IDE Status Register Primary (B0:D31:F2) 467
8-74 Offset 04h: Bus Master IDE Descriptor Table Pointer Register Primary (B0:D31:F2) ... 468
8-75 Offset 08h: Bus Master IDE Command Register Secondary (B0:D31:F2) 469
8-76 Offset 0Ah: Bus Master IDE Status Register Secondary (B0:D31:F2) 470

8-77 Offset 0Ch: Bus Master IDE Descriptor Table Pointer Register Secondary (B0:D31:F2)..... 471
8-78 Offset 10h: AHCI Index Register (B0:D31:F2) 471
8-79 Offset 14h: AHCI Index Data Register (B0:D31:F2) 472
8-80 Superset Registers 473
8-81 Offset SIDPBA + 00h: Serial ATA Index (B0:D31:F2) 473
8-82 Offset SIDPBA + 04h: Serial ATA Data (B0:D31:F2)..... 474
8-83 AHCI Register Address Map 474
8-84 AHCI Generic Host Control Registers..... 475
8-85 Offset 00h: Host Capabilities Register (B0:D31:F2)..... 475
8-86 Offset 04h: Global PCH Control Register (B0:D31:F2)..... 478
8-87 Offset 08h: Interrupt Status Register (B0:D31:F2) 479
8-88 Offset 0Ch: Ports Implemented Register (B0:D31:F2) 480
8-89 Offset 10h: Serial AHCI Version (B0:D31:F2) 480
8-90 Offset 14h: Serial Command Completion Coalescing Control Register (B0:D31:F2).... 481
8-91 Offset 18h: Serial Command Completion Coalescing Ports Register (B0:D31:F2)..... 482
8-92 Offset 1Ch: Serial Enclosure Management Location Register (B0:D31:F2) 482
8-93 Offset 20h: Serial Enclosure Management Control Register (B0:D31:F2)..... 483
8-94 Offset 24h: Serial Extended Host Capabilities (B0:D31:F2) 484
8-95 Offset SIDPBA + 04h: Serial ATA Index (B0:D31:F2) 485
8-96 Offset 70h: Serial AHCI Version (B0:D31:F2) 485
8-97 Offset A0h: Serial Vendor Specific (B0:D31:F2)..... 486
8-98 Port [4:5] DMA Register Address Map 487
8-99 Offset 300h: Port [4:5] Command List Base Address Register (B0:D31:F2)..... 488
8-100 Offset 304: Port [4:5] Command List Base Address Upper 32-Bits Register (B0:D31:F2).. 488
8-101 Offset 308h: Port [4:5] FIS Base Address Register (B0:D31:F2) 489
8-102 Offset 30Ch: Port [4:5] FIS Base Address Upper 32-Bits Register (B0:D31:F2)..... 489
8-103 Offset 310h: Port [4:5] Interrupt Status Register (B0:D31:F2)..... 490
8-104 Offset 314h: Port [4:5] Interrupt Enable Register (B0:D31:F2)..... 492
8-105 Offset 318h: Port [4:5] Command Register (B0:D31:F2) 494
8-106 Offset 320h: Port [4:5] Task File Data Register (B0:D31:F2)..... 497
8-107 Offset 324h: Port [4:5] Signature Register (B0:D31:F2)..... 498
8-108 Offset 328h: Port [4:5] Serial ATA Status Register (B0:D31:F2) 499
8-109 Offset 32Ch: Port [4:5] Serial ATA Control Register (B0:D31:F2)..... 501
8-110 Offset 330h: Port [4:5] Serial ATA Error Register (B0:D31:F2) 503
8-111 Offset 334h: Port [4:5] Serial ATA Active (B0:D31:F2)..... 504
8-112 Offset 338h: Port [4:5] Command Issue Register (B0:D31:F2)..... 505
9-1 SATA* Controller PCI Register Address Map 506
9-2 Offset 00h: VID: Vendor Identification Register (SATA—B0:D31:F5) 507
9-3 Offset 02h: Device Identification Register (SATA—B0:D31:F5) 508
9-4 Offset 04h: PCI Command Register (SATA—B0:D31:F5)..... 508
9-5 Offset 06h: PCI Status Register (SATA—B0:D31:F5) 509
9-6 Offset 08h: RID—Revision Identification Register (SATA—B0:D31:F5) 510
9-7 Offset 09h: Programming Interface Register (SATA—B0:D31:F5)..... 511
9-8 Offset 0Ah: Sub Class Code Register (SATA—B0:D31:F5) 511
9-9 Offset 0Bh: BCC—Base Class Code Register (SATA–B0:D31:F5) 512
9-10 Offset 0Dh: Primary Master Latency Timer Register (SATA–B0:D31:F5)..... 512
9-11 Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F5) 513
9-12 Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F5) 514
9-13 Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1) 514
9-14 Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F5) 515
9-15 Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F5) 515
9-16 Offset 24h: SATA Index/Data Pair Base Address Register (SATA–B0:D31:F5)..... 516
9-17 Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F5) 516
9-18 Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F5) 517
9-19 Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F5) 517
9-20 Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F5) 517
9-21 Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F5)..... 518

9-22 Offset 40h: IDE Timing Register (SATA–B0:D31:F5) 518
9-23 Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F5) 519
9-24 Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F5) 519
9-25 Offset 4Ah: IDE I/O Configuration Register (SATA–B0:D31:F5) 520
9-26 Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F5) 521
9-27 Offset 72h: PC—PCI Power Management Capabilities Register (SATA–B0:D31:F5) 521
9-28 Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F5) . 522
9-29 Offset 90h: MAP—Address Map Register (SATA–B0:D31:F5) 523
9-30 Offset 92h: Port Control and Status Register (SATA–B0:D31:F5) 524
9-31 Offset A8h: SATA Capability Register 0 (SATA–B0:D31:F5) 525
9-32 Offset ACh: SATA* Capability Register 1 (SATA–B0:D31:F5) 525
9-33 Offset B0h: FLR Capability ID (SATA–B0:D31:F5) 526
9-34 Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5) 526
9-35 Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5) 527
9-36 Offset B4h: FLR Control (SATA–B0:D31:F5) 527
9-37 Offset C0h: APM Trapping Control Register (SATA–B0:D31:F5) 528
9-38 Offset C4h: APM Trapping Control Register (SATA–B0:D31:F5) 528
9-39 Bus Master IDE I/O Registers (B0:D31:F5) 529
9-40 Offset 00h: Bus Master IDE Command Register (B0:D31:F5) 529
9-41 Offset 02h: Bus Master IDE Status Register (B0:D31:F5) 531
9-42 Offset 04h: Bus Master IDE Descriptor Table Pointer Register (B0:D31:F5) 532
9-43 Serial ATA Index/Data Pair Superset Registers 532
9-44 Offset 00h: SINDX—SATA* Index Register (B0:D31:F5) 533
9-45 Offset 04h: SDATA—SATA* Index Data Register (B0:D31:F5) 533
9-46 Offset 04h: PxSSTS—Serial ATA Status Register (B0:D31:F5) 534
9-47 Offset 04h: PxSCTL—Serial ATA Control Register (B0:D31:F5) 536
9-48 Offset 04h: PxSERR—Serial ATA Error Register (B0:D31:F5) 538
10-1 USB* EHCI PCI Register Address Map 540
10-2 Offset 00h: VID: Vendor Identification Register (USB EHCI—B0:D29:F0) 543
10-3 Offset 02h: DID—Device Identification Register (USB EHCI—B0:D29:F0) 543
10-4 Offset 04h: PCI Command Register (USB EHCI—B0:D29:F0) 544
10-5 Offset 06h: PCI Status Register (USB EHCI—B0:D29:F0) 545
10-6 Offset 08h: RID—Revision Identification Register (USB EHCI—B0:D29:F0) 546
10-7 Offset 09h: Programming Interface Register (USB EHCI—B0:D29:F0) 547
10-8 Offset 0Ah: Sub Class Code Register (USB EHCI—B0:D29:F0) 547
10-9 Offset 0Bh: BCC—Base Class Code Register (USB EHCI—B0:D29:F0) 547
10-10 Offset 0Dh: Primary Master Latency Timer Register (USB EHCI—B0:D29:F0) 548
10-11 Offset 0Eh: Header Type Register (USB EHCI—B0:D29:F0) 549
10-12 Offset 10h: Memory Base Address Register (USB EHCI—B0:D29:F0) 549
10-13 Offset 2Ch: Subsystem Vendor ID Register (USB EHCI—B0:D29:F0) 550
10-14 Offset 2Eh: Subsystem ID Register (USB EHCI—B0:D29:F0) 550
10-15 Offset 34h: Capabilities Pointer Register (USB EHCI—B0:D29:F0) 551
10-16 Offset 3Ch: Interrupt Line Register (USB EHCI—B0:D29:F0) 551
10-17 Offset 3Dh: Interrupt Pin Register (USB EHCI—B0:D29:F0) 551
10-18 Offset 50h: PCI Power Management Capability ID Register (USB EHCI—B0:D29:F0) .. 552
10-19 Offset 51h: Next Item Pointer #1 Register (USB EHCI—B0:D29:F0) 552
10-20 Offset 52h: Power Management Capabilities Register (USB EHCI—B0:D29:F0) 553
10-21 Offset 54h: Power Management Control/Status Register (USB EHCI—B0:D29:F0) 554
10-22 Offset 58h: Debug Port Capability ID Register (USB EHCI—B0:D29:F0) 555
10-23 Offset 59h: Next Item Pointer #2 Register (USB EHCI—B0:D29:F0) 555
10-24 Offset 5Ah: Debug Port Base Offset Register (USB EHCI—B0:D29:F0) 555
10-25 Offset 60h: USB* Release Number Register (USB EHCI—B0:D29:F0) 556
10-26 Offset 61h: Frame Length Adjustment Register (USB EHCI—B0:D29:F0) 557
10-27 Offset 62h: Port Wake Capability Register (USB EHCI—B0:D29:F0) 558
10-28 Offset 68h: Legacy Support Extended Capability Register (USB EHCI—B0:D29:F0) 559
10-29 Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0) . 560

10-30 Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register (USB EHCI—B0:D29:F0) 563
10-31 Offset 80h: ACCESS_CNTL—Access Control Register (USB EHCI—B0:D29:F0) 565
10-32 Offset 84h: EHCIIR1—EHCI Initialization Register 1 (USB EHCI—B0:D29:F0) 565
10-33 Offset 98h: FLR_CID—Function Level Reset Capability ID (USB EHCI—B0:D29:F0).... 566
10-34 Offset 99h: FLR_NEXT—Function Level Reset Next Capability Pointer (USB EHCI—B0:D29:F0) 566
10-35 Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0) 567
10-36 Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0) 567
10-37 Offset 9Ch: FLR_CTRL—Function Level Reset Control Register (USB EHCI—B0:D29:F0) ... 568
10-38 Offset 9Dh: FLR_STS—Function Level Reset Status Register (USB EHCI—B0:D29:F0) 568
10-39 Enhanced Host Controller Capability Registers 569
10-40 Offset 00h: CAPLENGTH—Capability Registers Length (USB EHCI—B0:D29:F0)..... 570
10-41 Offset 02h: HCIVERSION—Host Controller Interface Version Number Register (USB EHCI—B0:D29:F0) 570
10-42 Offset 04h: HCSPARAMS—Host Controller Structural Parameters (USB EHCI—B0:D29:F0) 571
10-43 Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB EHCI—B0:D29:F0) 572
10-44 Enhanced Host Controller Operational Register Address Map 573
10-45 Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0) 574
10-46 Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0) 577
10-47 Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)... 580
10-48 Offset 2Ch: FRINDEX—Frame Index Register (USB EHCI—B0:D29:F0) 582
10-49 Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment Register (USB EHCI—B0:D29:F0) 583
10-50 Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register (USB EHCI—B0:D29:F0) 584
10-51 Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register (USB EHCI—B0:D29:F0) 585
10-52 Offset 60h: CONFIGFLAG—Configure Flag Register (USB EHCI—B0:D29:F0)..... 586
10-53 Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) 587
10-54 Debug Port Register Address Map 592
10-55 Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0) 593
10-56 Offset A4h: USBPID—USB PIDs Register (USB EHCI—B0:D29:F0)..... 595
10-57 Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register (USB EHCI—B0:D29:F0)..... 596
10-58 Offset B0h: CONFIG—Configuration Register (USB EHCI—B0:D29:F0) 596
11-1 SMBus Controller PCI Register Address Map 597
11-2 Offset 00h: Vendor Identification Register (SMBus—B0:D31:F3) 598
11-3 Offset 02h: Device Identification Register (SMBus—B0:D31:F3) 598
11-4 Offset 04h: PCI Command Register (SMBus—B0:D31:F3) 599
11-5 Offset 06h: PCI Status Register (SMBus—B0:D31:F3) 600
11-6 Offset 08h: Revision Identification Register (SMBus—B0:D31:F3) 601
11-7 Offset 09h: Programming Interface Register (SMBus—B0:D31:F3)..... 601
11-8 Offset 0Ah: Sub Class Code Register (SMBus—B0:D31:F3) 601
11-9 Offset 0Bh: Base Class Code Register (SMBus—B0:D31:F3) 602
11-10 Offset 10h: SMBus Memory Base Address 0 (SMBus—B0:D31:F3)..... 602
11-11 Offset 14h: SMBus Memory Base Address 1 (SMBus—B0:D31:F3)..... 603
11-12 Offset 20h: SMBus Base Address Register (SMBus—B0:D31:F3) 603
11-13 Offset 2Ch: Subsystem Vendor Identification Register (SMBus—B0:D31:F3) 604
11-14 Offset 2Eh: Subsystem Identification Register (SMBus—B0:D31:F3) 604
11-15 Offset 3Ch: Interrupt Line Register (SMBus—B0:D31:F3) 605
11-16 Offset 3Dh: Interrupt Pin Register (SMBus—B0:D31:F3)..... 605
11-17 Offset 40h: Host Configuration Register (SMBus—B0:D31:F3) 606

11-18 SMBus Bus Registers....607
11-19 Offset 00h: Host Status Register (SMBus—B0:D31:F3)....608
11-20 Offset 02h: Host Control Register (SMBus—B0:D31:F3)....610
11-21 Offset 03h: Host Command Register (SMBus—B0:D31:F3)....612
11-22 Offset 04h: Transmit Slave Address Register (SMBus—B0:D31:F3)....612
11-23 Offset 05h: Host Data 0 Register (SMBus—B0:D31:F3)....613
11-24 Offset 06h: Host Data 1 Register (SMBus—B0:D31:F3)....613
11-25 Offset 07h: Host Block Data Byte Register (SMBus—B0:D31:F3)....614
11-26 Offset 08h: Packet Error Check (PEC) Register (SMBus—B0:D31:F3)....615
11-27 Offset 09h: Receive Slave Address Register (SMBus—B0:D31:F3)....615
11-28 Offset 0Ah: Receive Slave Data Register (SMBus—B0:D31:F3)....616
11-29 Offset 0Ch: Auxiliary Status Register (SMBus—B0:D31:F3)....616
11-30 Offset 0Dh: Auxiliary Control Register (SMBus—B0:D31:F3)....617
11-31 Offset 0Eh: SMLink Pin Control Register (SMBus—B0:D31:F3)....618
11-32 Offset 0Fh: SMBus Pin Control Register (SMBus—B0:D31:F3)....619
11-33 Offset 10h: Slave Status Register (SMBus—B0:D31:F3)....620
11-34 Offset 11h: Slave Command Register (SMBus—B0:D31:F3)....621
11-35 Offset 14h: Notify Device Address Register (SMBus—B0:D31:F3)....622
11-36 Offset 16h: Notify Data Low Byte Register (SMBus—B0:D31:F3)....622
11-37 Offset 17h: Notify Data High Byte Register (SMBus—B0:D31:F3)....623
12-1 PCI Express* Configuration Registers Address Map....624
12-2 Offset 00h: Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)....627
12-3 Offset 02h: DID—Device Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3) 627
12-4 Offset 04h: PCI COMMAND Register (PCI Express*—B0:D28:F0/F1/F2/F3)....628
12-5 Offset 06h: PCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)....629
12-6 Offset 08h: RID—Revision Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3) ..631
12-7 Offset 09h: Programming Interface Register (PCI Express*—B0:D28:F0/F1/F2/F3)....631
12-8 Offset 0Ah: Sub Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)....632
12-9 Offset 0Bh: Base Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)....632
12-10 Offset 0Ch: Cache Line Size Register (PCI Express*—B0:D28:F0/F1/F2/F3)....633
12-11 Offset 0Dh: Primary Latency Timer Register (PCI Express*—B0:D28:F0/F1/F2/F3)....633
12-12 Offset 0Eh: Header Type Register (PCI Express*—B0:D28:F0/F1/F2/F3)....634
12-13 Offset 18h: Bus Number Register (PCI Express*—B0:D28:F0/F1/F2/F3)....634
12-14 Offset 1Bh: Secondary Latency Timer (PCI Express*—B0:D28:F0/F1/F2/F3)....635
12-15 Offset 1Ch: IOBL—I/O Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3) 635
12-16 Offset 1Eh: Secondary Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)....636
12-17 Offset 20h: Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)....637
12-18 Offset 24h: Prefetchable Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)....638
12-19 Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)....638
12-20 Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)....639
12-21 Offset 34h: Capabilities List Pointer Register (PCI Express*—B0:D28:F0/F1/F2/F3)....639
12-22 Offset 3Ch: Interrupt Information Register (PCI Express*—B0:D28:F0/F1/F2/F3)....640
12-23 Offset 3Eh: Bridge Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)....640
12-24 Offset 40h: Capabilities List Register (PCI Express*—B0:D28:F0/F1/F2/F3)....642
12-25 Offset 42h: PCI Express* Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)....642
12-26 Offset 44h: Device Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)....643
12-27 Offset 48h: Device Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)....644
12-28 Offset 4Ah: Device Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)....645
12-29 Offset 4Ch: Link Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)....646
12-30 Offset 50h: Link Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)....648
12-31 Offset 52h: Link Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)....649
12-32 Offset 54h: Slot Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)....650
12-33 Offset 58h: Slot Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)....651
12-34 Offset 5Ah: Slot Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)....652

12-35 Offset 5Ch: Root Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 653
12-36 Offset 60h: Root Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 654
12-37 Offset 64h: Device Capabilities 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3) 654
12-38 Offset 68h: Device Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3) 655
12-39 Offset 70h: Link Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 656
12-40 Offset 80h: Message Signaled Interrupt Identifiers Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 656
12-41 Offset 82h: Message Signaled Interrupt Message Control Register (PCI Express*—B0:D28:F0/F1/F2/F3) 657
12-42 Offset 84h: Message Signaled Interrupt Message Address Register (PCI Express*—B0:D28:F0/F1/F2/F3) 658
12-43 Offset 88h: Message Signaled Interrupt Message Data Register (PCI Express*—B0:D28:F0/F1/F2/F3) 658
12-44 Offset 90h: Subsystem Vendor Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3). 659
12-45 Offset 94h: Subsystem Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 659
12-46 Offset A0h: Power Management Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3) 660
12-47 Offset A2h: PCI Power Management Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3) 660
12-48 Offset A4h: PCI Power Management Control and Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) 661
12-49 Offset D4h: Miscellaneous Port Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)..... 662
12-50 Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 663
12-51 Offset DCh: SMI/SCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) 666
12-52 Offset E1h: Root Port Dynamic Clock Gating Enable (PCI Express*—B0:D28:F0/F1/F2/F3) 667
12-53 Offset E8h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3). 668
12-54 Offset 104h: Uncorrectable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)... 668
12-55 Offset 108h: Uncorrectable Error Mask (PCI Express*—B0:D28:F0/F1/F2/F3)..... 670
12-56 Offset 10Ch: Uncorrectable Error Severity (PCI Express*—B0:D28:F0/F1/F2/F3) 672
12-57 Offset 110h: Correctable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) 673
12-58 Offset 114h: Correctable Error Mask Register (PCI Express*—B0:D28:F0/F1/F2/F3).. 674
12-59 Offset 118h: Advanced Error Capabilities and Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 675
12-60 Offset 130h: Root Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) 676
12-61 Offset 180h: Root Complex Topology Capability List Register (PCI Express*—B0:D28:F0/F1/F2/F3)..... 677
12-62 Offset 184h: Element Self Description Register (PCI Express*—B0:D28:F0/F1/F2/F3) 677
12-63 Offset 190h: Upstream Link Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)... 678
12-64 Offset 198h: Upstream Link Base Address Register (PCI Express*—B0:D28:F0/F1/F2/F3) 679
12-65 Offset 300h: PCI Express* Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3) 679
12-66 Offset 318h: PCI Express* Extended Test Mode Register (PCI Express*—B0:D28:F0/F1/F2/F3) 680
12-67 OFfset 324h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3) 680
13-1 Memory Mapped Registers..... 681
13-2 Offset 00h: General Capabilities and Identification Register..... 682
13-3 Offset 010h: General Configuration Register 683
13-4 Offset 020h: General Interrupt Status Register..... 684
13-5 Offset 0F0h: Main Counter Value Register..... 685

13-6 Offset 100h: Timer n Configuration and Capabilities Register 687
13-7 Offset 108h: Timer n Comparator Value Register 691
14-1 Serial Peripheral Interface (SPI) Register Address Map 692
14-2 Offset 00h: BIOS Flash Primary Region Register (SPI Memory Mapped Configuration Registers) 694
14-3 Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) 695
14-4 Offset 06h: Hardware Sequencing Flash Control Register SPI Memory Mapped Configuration Registers) 697
14-5 Offset 08h: Flash Address Register (SPI Memory Mapped Configuration Registers) 698
14-6 Offset 14h: Flash Data [N] Register (SPI Memory Mapped Configuration Registers) ... 699
14-7 Offset 50h: Flash Regions Access Permissions Register (SPI Memory Mapped Configuration Registers) 700
14-8 Offset 54h: Flash Region 0 (Flash Descriptor) Register (SPI Memory Mapped Configuration Registers) 701
14-9 Offset 58h: Offset 00h: Flash Region 1 (BIOS Descriptor) Register (SPI Memory Mapped Configuration Registers) 701
14-10 Offset 5Ch: Flash Region 2 (Intel ME) Register (SPI Memory Mapped Configuration Registers) 702
14-11 Offset 60h: Flash Region 3 Register (SPI Memory Mapped Configuration Registers) ... 702
14-12 Offset 64h: Flash Region 4 (Platform Data) Register (SPI Memory Mapped Configuration Registers) 703
14-13 Offset 74h: Protected Range 0 Register (SPI Memory Mapped Configuration Registers) ... 703
14-14 Offset 78h: Protected Range 1 Register (SPI Memory Mapped Configuration Registers) ... 704
14-15 Offset 7Ch: Protected Range 2 Register (SPI Memory Mapped Configuration Registers) ... 705
14-16 Offset 80h: Protected Range 3 Register (SPI Memory Mapped Configuration Registers) ... 706
14-17 Offset 84h: Protected Range 4 Register (SPI Memory Mapped Configuration Registers) ... 707
14-18 Offset 90h: Software Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) 708
14-19 Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers) 709
14-20 Offset 94h: Prefix Opcode Configuration Register (SPI Memory Mapped Configuration Registers) 711
14-21 Offset 96h: Opcode Type Configuration Register (SPI Memory Mapped Configuration Registers) 712
14-22 Offset 98h: Opcode Menu Configuration Register (SPI Memory Mapped Configuration Registers) 713
14-23 Offset A0h: BIOS Base Address Configuration Register (SPI Memory Mapped Configuration Registers) 714
14-24 Offset B0h: Flash Descriptor Observability Control Register (SPI Memory Mapped Configuration Registers) 715
14-25 Offset B4h: Flash Descriptor Observability Data Register SPI Memory Mapped Configuration Registers) 715
14-26 Offset C0h: Additional Flash Control Register (SPI Memory Mapped Configuration Registers) 716
14-27 Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) 717
14-28 Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) 719
14-29 Offset D0h: Flash Partition Boundary (SPI Memory Mapped Configuration Registers).. 721
15-1 Summary of UART Registers in I/O Space (DLAB=0) 722
15-2 Summary of UART Registers in I/O Space (DLAB=1) 722
15-3 Summary of UART Timer Registers in I/O Space 723
15-4 Internal Register Descriptions 723

15-5 Offset 00h: RBR—Receive Buffer Register 724
15-6 Offset 00h: THR—Transmit Holding Register 724
15-7 Offset 01h: IER—Interrupt Enable Register 725
15-8 Interrupt Conditions 726
15-9 Offset 02h: IIR—Interrupt Identification Register 726
15-10 Interrupt Identification Register Decode 727
15-11 Offset 02h: FCR—FIFO Control Register 728
15-12 Offset 03h: LCR—Line Control Register 729
15-13 Offset 04h: MCR—Modem Control Register 731
15-14 Offset 05h: LSR—Line Status Register 733
15-15 Offset 06h: MSR—Modem Status Register 736
15-16 Offset 07h: SCR—Scratchpad Register 737
15-17 Offset 00h: DLL—Programmable Baud Rate Generator Divisor Latch Register Low ... 737
15-18 Offset 01h: DLH—Programmable Baud Rate Generator Divisor Latch Register High ... 738
15-19 Summary of Watchdog Timer Registers in I/O Space 739
15-20 Offset 00h: PV1R0—Preload Value 1 Register 0 739
15-21 Offset 01h: PV1R1—Preload Value 1 Register 1 740
15-22 Offset 02h: PV1R2—Preload Value 1 Register 2 740
15-23 Offset 04h: PV2R0—Preload Value 2 Register 0 741
15-24 Offset 05h: PV2R1—Preload Value 2 Register 1 741
15-25 Offset 06h: PV2R2—Preload Value 2 Register 2 742
15-26 Offset 08h: GISR—General Interrupt Status Register 742
15-27 Offset 0Ch: RR0—Reload Register 0 743
15-28 Offset 0Dh: RR1—Reload Register 1 743
15-29 Offset 10h: WDTCR—WDT Configuration Register 744
15-30 Offset 18h: WDTLR—WDT Lock Register 745
15-31 Configuration Register Summary 746
15-32 Logical Device 4 (Serial Port 1) 749
15-33 Logical Device 5 (Serial Port 2) 750
15-34 Logical Device 6 (Watch Dog Timer) 752
16-1 PCI Header (B0:D31:F7) 754
16-2 Offset 00h: ID—Identifiers 754
16-3 Offset 04h: CMD—Command 755
16-4 Offset 06h: DSTS—Device Status 756
16-5 Offset 08h: RID—Revision ID 757
16-6 Offset 09h: CC—Class Code 757
16-7 Offset 0Ch: CLS—Cache Line Size 758
16-8 Offset 0Dh: MLT—Master Latency Timer 758
16-9 Offset 0Eh: HTYPE—Header Type 758
16-10 Offset 10h: TBAR—WDT Table Base Address 759
16-11 Offset 14h: PBAR—WDT PBA Base Address 759
16-12 Offset 18h: CBAR—WDT CFG Base Address 760
16-13 Offset 2Ch: SSVID—Subsystem Vendor ID 760
16-14 Offset 2Eh: SSID Subsystem ID 761
16-15 Offset 34h: CAP—Capabilities Pointer 761
16-16 PCI MSI-X Capability 761
16-17 Offset 80h: MID—MSI-X Capability ID 762
16-18 Offset 82h: MC—MSI-X Control & Status 762
16-19 Offset 84h: MT—MSI-X Table 762
16-20 Offset 88h: MP—MSI-X Message PBA 763
16-21 Table BAR Range 764
16-22 Offset 00h: WDT0MAW—WDT 0 Msg Add WARN 765
16-23 Offset 08h: WDT0MDW—WDT 0 Msg Data WARN 766
16-24 Offset 0Ch: WDT0VCW—WDT 0 Vector Ctrl WARN 766
16-25 Offset 18h: WDT0MDR—WDT 0 Msg Data RESET 767
16-26 Offset 1Ch: WDT0VCR—WDT 0 Vector Ctrl RESET 767
16-27 PBA BAR Range 768
16-28 Offset 00h: WDT_PBA0—Pending Bit Array 0 769
16-29 Offset 04h: WDT_PBA1—Pending Bit Array 1 770

16-30 Offset 08h: WDT_PBA2—Pending Bit Array 2 771
16-31 Offset 0Ch: WDT_PBA3—Pending Bit Array 3 772
16-32 CFG BAR Range 773
16-33 Offset 00h: WDT0COUNT—WDT Count Register 773
16-34 Offset 04h: WDT0CMD—WDT Cmd/Status Register 774
16-35 Offset 200h: WDT_PSCALE 774
16-36 Offset 204h: WDT_GBLCFG 775
17-1 Bus 0, Device 31, Function 6: Summary of Thermal Reporting PCI Configuration Registers 777
17-2 Offset 00h: ID: Vendor Identification Register 779
17-3 Offset 02h: DID—Device Identification Register 779
17-4 Offset 04h: ID—Command Register 780
17-5 Offset 06h: ID—STS—Status Register 781
17-6 Offset 08h: RID—Revision Identification Register 782
17-7 Offset 09h: PI—Programming Interface Register 782
17-8 Offset 0Ah: SCC—Sub Class Code Register 782
17-9 Offset 0Bh: BCC—Base Class Code Register 783
17-10 Offset 0Ch: Cache Line Size 783
17-11 Offset 0Dh: Latency Timer 783
17-12 Offset 0Eh: Header Type 784
17-13 Offset 10h: Thermal Base 784
17-14 Offset 14h: TBARH—Thermal Base High DWord 785
17-15 Offset 2Ch: Subsystem Vendor ID 785
17-16 Offset 2Eh: Subsystem ID 786
17-17 Offset 34h: Capabilities Pointer 786
17-18 Offset 3Ch: Interrupt Line 787
17-19 Offset 3Dh: Interrupt Pin 787
17-20 Offset 40h: BIOS Assigned Thermal Base Address 788
17-21 Offset 44h: BIOS Assigned Thermal Base High DWord 788
17-22 Offset 50h: PCI Power Management Capability ID 789
17-23 Offset 52h: Power Management Capabilities 789
17-24 Offset 54h: Power Management Control And Status 790
17-25 Offset 80h: MID—Message Signaled Interrupt Identifiers 791
17-26 Offset 82h: MC—Message Signaled Interrupt Message Control 791
17-27 Offset 84h: MA—Message Signaled Interrupt Message Address 792
17-28 Offset 88h: MD—Message Signaled Interrupt Message Data 792
17-29 Offset 94h: CCTS0RD—Thermal Sensor 0 Remote Diode 792
17-30 Offset 96h: CCTS0SSI—Thermal Sensor 0 Sense Stage Inputs 793
17-31 Offset 98h: CCTS0DSAC0—Thermal Sensor 0 Delta Sigma ADC Control 0 794
17-32 Offset 9Ch: CCTS0DSAC1—Thermal Sensor 0 Delta Sigma ADC Control 1 794
17-33 Offset A0h: CCTS0C—Thermal Sensor 0 Calibration 795
17-34 Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0 795
17-35 Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1 797
17-36 Offset ACh: CCTS0BP—Thermal Sensor 0 Bandgap Pins 798
17-37 Offset B4h: CCTS1RD—Thermal Sensor 1 Remote Diode 798
17-38 Offset B6h: CCTS1SSI—Thermal Sensor 1 Sense Stage Inputs 799
17-39 Offset B8h: CCTS1DSAC0—Thermal Sensor 1 Delta Sigma ADC Control 0 799
17-40 Offset BCh: CCTS1DSAC1—Thermal Sensor 1 Delta Sigma ADC Control 1 800
17-41 Offset C0h: CCTS1C—Thermal Sensor 1 Calibration 800
17-42 Offset C4h: CCTS1TL0—Thermal Sensor 1 Top Logic 0 801
17-43 Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1 802
17-44 Offset CCh: CCTS1BP—Thermal Sensor 1 Bandgap Pins 803
17-45 Offset F8h: MANID—Manufacturing/Process 803
17-46 Bus 0, Device 31, Function 6: Summary of Thermal Reporting Registers Mapped Through TBARB BAR 804
17-47 Offset 00h: TS0IU—Thermal Sensor 0 In Use 806
17-48 Offset 01h: TS0C—Thermal Sensor 0 Control 807
17-49 Offset 02h: TS0S—Thermal Sensor 0 Status 809
17-50 Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read 810

17-51 Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point 810
17-52 Offset 08h: TS0CO—Thermal Sensor 0 Calibration Offset 812
17-53 Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd 812
17-54 Offset 0Ch: TS0ES—Thermal Sensor 0 Error Status 813
17-55 Offset 0Dh: TS0GPEN—Thermal Sensor 0 General Purpose Event Enables 814
17-56 Offset 0Eh: TS0PC—Thermal Sensor 0 Policy Control 815
17-57 Offset 10h: CPEC—CPU Power Error Correction Data 816
17-58 Offset 12h: C0TA—CPU0 Temperature Adjust 816
17-59 Offset 16h: C1TA—CPU1Temperature Adjust 817
17-60 Offset 1Ah: MC—MPC Control 817
17-61 Offset 22h: Timestamp 818
17-62 Offset 24h: DIMMEN—DIMM Enable 819
17-63 Offset 30h: C0TV—CPU0 Temperature Value 820
17-64 Offset 32h: C1TV—CPU1 Temperature Value 821
17-65 Offset 34h: C0EV—CPU0 Energy Value 821
17-66 Offset 38h: C1EV—CPU1 Energy Value 821
17-67 Offset 3Fh: AE—Alert Enable 822
17-68 Offset 40h: TS1IU—Thermal Sensor 1 In Use 823
17-69 Offset 41h: TS1C—Thermal Sensor 1 Control 824
17-70 Offset 42h: TS1S—Thermal Sensor 1 Status 825
17-71 Offset 43h: TS1TR—Thermal Sensor 1 Thermometer Read 826
17-72 Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point 826
17-73 Offset 48h: TS1CO—Thermal Sensor 1 Calibration Offset 828
17-74 Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd 829
17-75 Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status 829
17-76 Offset 4Dh: TS1GPEN—Thermal Sensor 1 General Purpose Event Enables 831
17-77 Offset 4Eh: TS1PC—Thermal Sensor 1 Policy Control 832
17-78 Offset 50h: HTS—HOST Turbo Status 833
17-79 Offset 56h: MTL—MCP Temperature Limit 833
17-80 Offset 58h: MTV—MCH Temperature Value 834
17-81 Offset 60h: MCPTV—MCP Temperature Value 834
17-82 Offset 64h: MMPC—Max MCH Power Clamp 834
17-83 Offset 66h: MMCPPC—Max MCP Power Clamp 835
17-84 Offset 82h: TS0PIEN—Thermal Sensor 0 PCI Interrupt Event Enables 835
17-85 Offset 83h: TS0LOCK—Thermal Sensor 0 Register Lock Controls 836
17-86 Offset 98h: STS—SMBus Turbo Status 836
17-87 Offset 9Ch: SEC—SMBus Event Clear 837
17-88 Offset A4h: TC3—Thermal Compares 3 837
17-89 Offset A8h: TC1—Thermal Compares 1 838
17-90 Offset ACh: TC2—Thermal Compares 2 838
17-91 Offset B0h: DIMM0—CPU0 DIMM Data 839
17-92 Offset B4h: DIMM1—CPU1 DIMM Data 841
17-93 Offset B8h: DIMMID—DIMM ID 842
17-94 Offset BCh: ECPCLAMP—EC Power Clamp Data 844
17-95 Offset C2h: TS1PIEN—Thermal Sensor 1PCI Interrupt Event Enables 844
17-96 Offset C3h: TS1LOCK—Thermal Sensor 1 Register Lock Controls 845
17-97 Offset D8h: ITV—Internal Temperature Values 845
18-1 MEI1 Configuration Registers Address Map (MEI1—B0:D22:F0) 846
18-1 MEI1 MMIO Register Address Map 857
18-2 MEI2 Configuration Registers Address Map (MEI2—B0:D22:F1) 860
18-3 MEI2 MMIO Register Address Map 869
19-1 PCIe* Commands Supported by RI 877
19-2 PCIe* EP Function Mapping 877
19-3 EP Mapping of BARs to MMIOs 878
19-4 PCIe* Error Sources 880
19-5 Advisory Error Cases 881
19-6 Device Specific Errors 884
19-7 Supported PCIe* Transactions 885
19-8 PCIe* Request Header Format for 32-bit Memory Addressing 886

19-9 PCIe* Request Header Format for 64-bit Memory Addressing....886
19-10 PCIe* Completion Header Format....886
19-11 PCI Completion Request Generation....887
19-12 PCIe* Request Header Format for Messages....888
19-13 PCIe* Message Generation....888
20-1 EP PCI Configuration Registers....892
20-2 Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Configuration Registers 894
20-3 Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Registers Mapped Through PMISCRBAR+1A000h Memory BAR....897
20-4 Bus M, Function 0: Summary of PCIe Intel® QuickAssist Registers Mapped Through PMISCSBAR+1A000h Memory BAR....897
20-5 Bus M, Function 8+Index 1: Summary of PCIe Intel® QuickAssist Configuration Registers 897
20-6 PVID—PF Vendor Identification Register....899
20-7 PDID—PF Device Identification Register....899
20-8 PPCICMD—PF Device Command Register....900
20-9 PPCISTS—PF PCI Device Status Register....901
20-10 PRID—PF Revision ID Register....903
20-11 PCC—PF Class Code Register....903
20-12 PHDR—PF Header Type Register....904
20-13 PeSRAMLBAR—PF Memory Lower Base Address Register....904
20-14 PeQATUBAR—PF QAT Upper Base Address Register....905
20-15 PMISCLBAR—PF Miscellaneous Lower Base Address Register....905
20-16 PMISCUBAR—PF Miscellaneous Upper Base Address Register....906
20-17 PETRINGCSRLBAR—PF Ring CSR Lower Base Address Register....906
20-18 PSVID—PF Subsystem Vendor ID Register....907
20-19 PSID—PF Subsystem ID Register....907
20-20 PCP—PF Capabilities Pointer Register....908
20-21 PIRQL—PF Interrupt Line Register....908
20-22 PIRQP—PF Interrupt Pin Register....909
20-23 SKU Register....909
20-24 SKU Register 2....910
20-25 PMSI-X—PF Message Signalled Interrupt X Capability ID Register....911
20-26 PMSIXNCP—PF MSIX Next Capability Pointer Register....911
20-27 PMSIXCNTL—PF Message Signalled Interrupt X Control Register....912
20-28 PMSIXTBIR—PF MSI-X Table Offset & Table BIR Register....912
20-29 PMSIXPBABIR—PF MSI-X Pending Bit Array & BIR Offset Register....913
20-30 PPMCAP—PF Power Management Capabilities ID Register....913
20-31 PPMCP—PF Power Management Next Capability Pointer Register....914
20-32 PPMC—PF Power Management Capabilities Register....914
20-33 PPMCSR—PF Power Management Control and Status Register....915
20-34 PPCID—PF PCI Express Capability Register....917
20-35 PPCP—PF PCI Express Next Capability Pointer Register....917
20-36 PPCR—PF PCI Express Capabilities Register....918
20-37 PPDCAP—PF PCI Express Device Capabilities Register....918
20-38 PPDCNTL—PF PCI Express Device Control Register....920
20-39 PPDSTAT—PF PCI Express Device Status Register....921
20-40 PLCAPR—PF Link Capabilities Register....922
20-41 PLCNTLR—PF Link Control Register....924
20-42 PLSR—PF Link Status Register....926
20-43 PDCAPR2—PF Device Capabilities 2 Register....927
20-44 PDCNTR2—PF Device Control 2 Register....928
20-45 PLCNTLR2—PF Link Control 2 Register....929
20-46 PLSR2—PF Link Status 2 Register....931
20-47 PMSICID—PF Message Signalled Interrupt Capability ID Register....931
20-48 PMSINCP—PF Message Signalled Interrupt Next Capability Pointer Register....932
20-49 PMSICTL—PF Message Signalled Interrupt Control Register....932
20-50 PMSILADDR—PF Message Signalled Interrupt Lower Address Register....933

20-51 PMSIUADDR—PF Message Signalled Interrupt Upper Address Register 933
20-52 PMSIDATA—PF Message Signalled Interrupt Data Register 934
20-53 PMSIMSK - PF Message Signalled Interrupt Mask Register 934
20-54 PMSIPND—PF Message Signalled Interrupt Pending Register 935
20-55 PPCIEAERCAPID—PF PCI Express AER Capability ID Register 935
20-56 PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register 936
20-57 PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register 937
20-58 PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register 938
20-59 PPAERCS—PF PCI Express AER Correctable Error Register 939
20-60 PPAERCM—PF PCI Express AER Correctable Error Mask Register 940
20-61 PPAERCTLCAP—PF PCI Express AER Control and Capability Register 941
20-62 PPAERHDRLOG0—PF PCI Express AER Header Log 0 Register 941
20-63 PPAERHDRLOG1—PF PCI Express AER Header Log 1 Register 942
20-64 PPAERHDRLOG2—PF PCI Express AER Header Log 2 Register 942
20-65 PPAERHDRLOG3—PF PCI Express AER Header Log 3 Register 943
20-66 PARIDHDR—PF Alternative Routing ID Capability Header 944
20-67 PFARICAP—PF ARI Capabilities Register 944
20-68 PARIDCTL—PF Alternative Routing ID Control Register 945
20-69 PSRIOVCAPID—PF SR IOV Capability ID Register 945
20-70 PSRIOVCVNC—PF SRIOV Capability Version and Next Capability Pointer Register 946
20-71 PSRIOVCAP—PF SRIOV Capabilities Register 947
20-72 PSRIOVCS—PF SRIOV Control and Status Register 947
20-73 PSRIOVMTOTINI—PF SRIOV Initial and Total VFs Register 948
20-74 PSRIOVNUMVF—PF SRIOV Number of VFs Register 948
20-75 PSRIOVFVFO—PF SRIOV First VF Offset Register 949
20-76 PSRIOVVFS—PF SRIOV VF Stride Register 949
20-77 PSRIOVFDID—PF SRIOV VF Device ID Register 950
20-78 PSRIOVPAGESIZE—PF SRIOV Supported Page Size Register 950
20-79 PSRIOVSYSPS—PF SRIOV System Page Size Register 951
20-80 PSRIOVLBAR0—SRIOV Lower BAR0 Register 952
20-81 PSRIOVUBAR0—SRIOV Upper BAR0 Register 953
20-82 PSRIOVLBAR1—SRIOV Lower BAR1 Register 954
20-83 PSRIOVUBAR1—SRIOV Upper BAR1 Register 955
20-84 PSRIOVVFMA—PF SRIOV VF Migration Array Register 955
20-85 VVID[0:15]—VF Vendor Identification Register 956
20-86 VDID[0:15]—VF Device Identification Register 956
20-87 VPCICMD[0:15]—VF Device Command Register 957
20-88 VPCISTS[0:15]—VF PCI Device Status Register 958
20-89 VRID[0:15]—VF Revision ID Register 959
20-90 VCC[0:15]—VF Class Code Register 959
20-91 VHDR[0:15]—VF Header Type Register 959
20-92 VSVID[0:15]—VF Subsystem Vendor ID Register 960
20-93 VSID[0:15]—VF Subsystem ID Register 960
20-94 VCP[0:15]—VF Capabilities Pointer Register 961
20-95 VIRQL[0:15]—VF Interrupt Line Register 961
20-96 VIRQP[0:15]—VF Interrupt Pin Register 961
20-97 VPCID[0:15]—VF PCI Express Capability ID Register 962
20-98 VPCP[0:15]—VF PCI Express Next Capability Pointer Register 962
20-99 VPCR[0:15]—VF PCI Express Capabilities Register 963
20-100 VPDCAP[0:15]—VF PCI Express Device Capabilities Register 963
20-101 VPDC[0:15]—VF PCI Express Device Control Register 964
20-102 VPDS[0:15]—VF PCI Express Device Status Register 965
20-103 VLCR[0:15]—VF Link Capabilities Register 966
20-104 VLCNTLR[0:15]—VF Link Control Register 968
20-105 VLSR[0:15]—VF Link Status Register 969
20-106 DCAPR2[0:15]—Device Capabilities 2 Register 969
20-107 VMSICID[0:15]—Message Signalled Interrupt Capability ID Register 970
20-108 VMSINCP[0:15]—Message Signalled Interrupt Next Capability Pointer Register 970
20-109 VMSICTL[0:15]—Message Signalled Interrupt Control Register 971

20-110 VMSILADDR[0:15]—Message Signalled Interrupt Lower Address Register 971
20-111 VMSIUADDR[0:15]—Message Signalled Interrupt Upper Address Register 972
20-112 VMSIDATA[0:15]—Message Signalled Interrupt Data Register 972
20-113 VMSIMSK—VF Message Signalled Interrupt Mask Register 972
20-114 VMSIPND—VF Message Signalled Interrupt Pending Register 973
20-115 VPCIEAERCAPID[0:15]—VF PCI Express AER Capability ID Register 973
20-116 VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register 974
20-117 VPAERUCM[0:15]—VF PCI Express AER Uncorrectable Error Mask Register 975
20-118 VPAERUCSEV[0:15]—VF PCI Express AER Uncorrectable Error Severity Register 977
20-119 VPAERCS[0:15]—VF PCI Express AER Correctable Error Status Register 978
20-120 VPAERCM[0:15]—VF PCI Express AER Correctable Error Mask Register 979
20-121 VPAERCTLCAP[0:15]—VF PCI Express AER Control and Capability Register 980
20-122 VPAERHDRLOG0[0:15]—VF PCI Express AER Header Log 0 Register 981
20-123 VPAERHDRLOG1[0:15]—VF PCI Express AER Header Log 1 Register 982
20-124 VPAERHDRLOG2[0:15]—VF PCI Express AER Header Log 2 Register 982
20-125 VPAERHDRLOG3[0:15]—VF PCI Express AER Header Log 3 Register 983
20-126 VARIDHDR[0:15]—VF Alternative Routing ID Capability Header 983
20-127 VFARICAP[0:15]—VF ARI Capabilities Register 984
20-128 VARIDCTL[0:15]— VF Alternative Routing ID Control Register 984
20-129 ERRSOU2—Error Source Register 2 985
20-130 ERRMSK2—Error Source Mask Register 2 985
20-131 SINTPF—Signal Raw PF Interrupt Register 986
20-132 SMIAPF—Signal IA PF Interrupt Mask Register 986
20-133 GBECFGMMIOV - GBE Configuration and MMIO Valid Register 987
20-134 SINTGBE[0:3]—Signal Raw PF Interrupt for GBE Register 988
20-135 SMIAGBE[0:3]—Signal IA PF Interrupt Mask GBE Register 988
21-1 Glossary 991
21-2 GbE Controller Features 996
21-3 GbE Controller Network Features 996
21-4 GbE Controller Host Interface Features 997
21-5 GbE Controller LAN Functions Features 997
21-6 GbE Controller LAN Performance Features 998
21-7 GbE Controller Virtualization Features 998
21-8 Transmit Data Flow 1002
21-9 Receive Data Flow 1003
22-1 LAN Traffic Attributes 1005
22-2 SMBus ARA Cycle Format 1008
22-3 Asynchronous Notify Command Format 1008
22-4 Direct Receive - LAN Packet Receive Transaction Format 1009
22-5 Direct Receive - Status Change Transaction Format 1009
22-6 Flags in Transmit Packet Transactions 1011
22-7 Unique Device Identifier (UDID) 1015
22-8 Dynamic and Persistent Address, PEC Support Bit 1015
22-9 Version/Revision: UDID Version 1, Silicon Revision 1015
22-10 Silicon Revision ID 1015
22-11 Vendor Specific ID 1016
22-12 EEPROM Structure 1016
22-13 EEPROM Auto-Load Sequence 1017
22-14 Notes on EEPROM Words 1022
22-15 Small Resource Structure 1022
22-16 Large Resource Structure 1023
22-17 VPD Structure 1023
22-18 Encoding on LINK_MODE Field for Each Mode 1025
22-19 Ordered Sets 1029
22-20 802.3z Advertised Base Page Mapping 1031
22-21 802.3X Packet Format 1035
22-22 Forwarding of PAUSE Packet to Host (DPF Bit) 1036
22-23 1036
23-1 Notes to Power-Up Timing Diagram 1041

23-2 GbE Controller Reset Effects - Common Resets....1046
23-3 GbE Controller Reset Effects - Per Function Resets....1047
24-1 EEPROM Top Level Partitioning....1059
24-2 Common and Lan Port 0 EEPROM Map....1059
24-3 LAN Ports 1, 2, and 3 EEPROM Map....1061
24-4 Supported MAC Connection Types....1063
24-5 CSR Auto Configuration Structure Format....1073
24-6 CSR Auto Configuration Power-Up Structure Format....1075
24-7 EICT Information Structure Format....1076
24-8 PHYAuto Configuration Structure Format....1077
24-9 HY Configuration Section Length....1077
24-10 Block CRC8 Structure....1077
24-11 PHY Number and PHY Address....1078
24-12 PHY Data....1078
24-13 PT LAN Configuration Structure Format....1082
24-14 Alternate MAC Address Block....1095
25-1 Power Up (Off to Dup to D0u to D0a)....1101
25-2 Transition From D0a to D3 and Back Without PCIE_EP_RST#....1102
25-3 Transition From D0a to D3 and Back With PCIE_EP_RST#....1103
25-4 Transition From D0a to Dr and Back Without Transition to D3....1104
25-5 Magic Packet Structure....1108
25-6 ARP Packet Structure and Processing....1109
25-7 IPv4 Packet Structure and Processing....1109
25-8 IPv6 Packet Structure and Processing....1110
25-9 IPX Diagnostic Responder Request Packet Structure and Processing....1111
25-10 IPX Packet Structure and Processing....1111
26-1 IPv4....1124
26-2 IPv6....1124
26-3 Legacy Receive Descriptor (RDESC) Layout....1131
26-4 Receive Status (RDESC.STATUS) Layout....1132
26-5 Receive Status Bits....1132
26-6 IPCS, L4CS, and UDPCS....1133
26-7 RXE, IPE and L4E....1133
26-8 VLAN Tag Field Layout (for 802.1q Packet)....1134
26-9 RDESC Descriptor Read Format....1134
26-10 RDESC Descriptor Write-Back Format....1135
26-11 RSS Type....1135
26-12 Packet Type LSB Bits (11:10)....1136
26-13 Receive Status (RDESC.STATUS) Layout of Last Descriptor....1137
26-14 Receive Status (RDESC.STATUS) Layout of Non-Last Descriptor....1137
26-15 Receive Errors (RDESC.ERRORS) Layout....1138
26-16 GBE Controller Split/Replicated Header Behavior....1144
26-17 Timestamp Layout in Buffer....1145
26-18 Supported Receive Checksum Capabilities....1147
26-19 Typical IPv6 Extended Header Format (Traditional Representation)....1148
26-20 Header Type Encoding and Lengths....1149
26-21 Descriptor Fields....1150
26-22 SCTP Header....1150
26-23 Transmit Descriptor (TDESC) Fetch Layout - Legacy Mode....1153
26-24 Transmit Descriptor (TDESC) Write-Back Layout - Legacy Mode....1153
26-25 Transmit Command (TDESC.CMD) Layout....1154
26-26 VLAN Tag Insertion Decision Table....1154
26-27 Transmit Status (TDESC.STA) Layout....1155
26-28 VLAN Field (TDESC.VLAN) Layout....1155
26-29 Transmit Context Descriptor (TDESC) Layout - (Type = 0010b)....1156
26-30 MACLEN Values....1156
26-31 Transmit Command (TDESC.TUCMD) Layout....1157
26-32 Valid Field in Context vs. Required Offload....1158
26-33 Advanced Transmit Data Descriptor (TDESD) Layout - (Type = 0011b)....1158

26-34 Advanced Tx Descriptor Write-back Format 1158
26-35 Transmit Data (TDESD.MAC) Layout 1159
26-36 Transmit Data (TDESD.DCMD) Layout 1159
26-37 Transmit Data (TDESD.POPTS) Layout 1160
26-38 Write Back Options For Large Send 1167
26-39 TCP/IP or UDP/IP Packet Format Sent by Host 1168
26-40 TCP/IP or UDP/IP Packet Format Sent by the Controller 1168
26-41 TCP Partial Pseudo-Header Sum for IPv4 1170
26-42 TCP Partial Pseudo-Header Sum for IPv6 1170
26-43 Supported Transmit Checksum Capabilities 1170
26-44 Conditions for Checksum Off Loading 1171
26-45 Checksum Per Packet Type 1176
26-46 Cause Allocation in the IVAR Registers - MSI and Legacy Mode 1179
26-47 Cause Allocation in the IVAR Registers 1180
26-48 Interrupt Registers - Legacy Mode 1180
26-49 Interrupt Registers - Extended Mode 1180
26-50 Settings for Different Interrupt Modes 1183
26-51 Comparing Packets 1189
26-52 TCI Bit Ordering 1190
26-53 Receive Processing in Double VLAN Mode 1192
26-54 Mode Encoding for LED Outputs 1193
26-55 Legacy DCA Systems 1195
26-56 DCA Systems 1195
26-57 Storm Control Interval by Speed 1205
26-58 Chronological Order of Events for Sync and Path Delay 1208
26-59 V1 and V2 PTP Message Structure 1213
26-60 PTP Message Over Layer 2 1215
26-61 PTP Message Over Layer 4 1215
26-62 Message Decoding for V1 (Control Field at Offset 32) 1215
26-63 Message Decoding for V2 (Message Id Field at Offset 0) 1215
26-64 IEEE 802.3 Mandatory Package Statistics 1216
26-65 IEEE 802.3 Recommended Package Statistics 1216
26-66 IEEE 802.3 Optional Package Statistics 1217
26-67 Microsoft* OID_GEN_STATISTICS 1217
26-68 RMON Statistics 1218
26-69 Linux net_device_stats 1219
27-1 SMBus Write Transactions 1223
27-2 TCO LAN Packet Status Data 1230
27-3 Packet Status Info 1230
27-5 Security errors 1230
27-6 Packet Type 1231
27-7 MNG Status 1231
27-4 Error Status Info 1231
27-8 Registers Written by the BMC 1240
27-9 Exclusive Traffic Behavior 1245
28-1 Address Space Regions 1247
28-2 IOADDR and IODATA in I/O Address Space 1248
28-3 IOADDR and IODATA in Configuration Address Space 1249
28-4 GbE Controller Register Field Attributes 1251
28-5 Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers 1252
28-6 Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR3[03]B:0:1+Index 1 Registers 1262
28-7 Register Summary 1263
28-8 MSI-X Register Summary 1265
28-9 Device Control Register—CTRL [0:3] (0x00000; R/W) 1266
28-10 Device Status Register— STATUS[0:3] (0x0008; R) 1269
28-11 Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W) 1271
28-12 Mappings for SDI Pins Used as GPI 1272

28-13 MDI Control Register—MDIC[0:3] (0x0020; R/W) 1273
28-14 MDC/MDIO Configuration Register—MDICNFG [0:3] (0x0E04; R/W) 1274
28-15 Copper/Fiber Switch Control—CONNSW[0:3] (0x0034; R/W)....1275
28-16 VLAN Ether Type—VET [0:3] (0x0038; R/W)....1275
28-17 LED Control—LEDCTL [0:3](0x0E00; RW)....1276
28-18 Internal Receive Packet Buffer Size—IRPBS [0:3] (0x2404; RO) 1277
28-19 Internal Transmit Packet Buffer Size—ITPBS [0:3] (0x3404; RO)....1278
28-20 EEPROM Control Register—EEC [0:3] (0x0010; R/W) 1279
28-21 EEPROM Read Register—EERD [0:3] (0x0014; RW) 1281
28-22 EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO)....1281
28-23 VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO)....1282
28-24 Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO) 1283
28-25 Management EEPROM Read/Write Data—EEMNGDATA [0:3] (0x1014; RO) 1284
28-26 Flow Control Address Low—FCAL [0:3] (0x0028; RO)....1285
28-27 Flow Control Address High—FCAH [0:3] (0x002C; RO) 1286
28-28 Flow Control Type—FCT [0:3] (0x0030; R/W) 1286
28-29 Flow Control Transmit Timer Value—FCTTV [0:3] (0x0170; R/W)....1287
28-30 Flow Control Receive Threshold Low—FCRTL0 [0:3] (0x2160; R/W) 1288
28-31 Flow Control Receive Threshold High—FCRTH0 [0:3] (0x2168; R/W)....1289
28-32 Flow Control Refresh Threshold Value—FCRTV[0:3](0x2460; R/W)....1290
28-33 Flow Control Status—FCSTS0 [0:3] (0x2464; RO) 1291
28-34 Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO) 1291
28-35 Mirrored Revision ID—MREVID[0:3] (0x5B64; R/W)....1293
28-36 PCIe* Control Extended Register—GCR_EXT[0:3] (0x5B6C; R/W)....1294
28-37 Software Semaphore—SWSM[0:3] (0x5B50; R/W) 1295
28-38 Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS) 1296
28-39 Firmware Synchronization—SW_FW_SYNC[0:3] (0x5B5C; RWS)....1298
28-40 Software Mailbox Write—SWMBWR[0:3] (0x5B04; R/W)....1299
28-41 Software Mailbox 0—SWMB0[0:3] (0x5B08; RO) 1299
28-42 Software Mailbox 1—SWMB1[0:3] (0x5B0C; RO)....1299
28-43 Software Mailbox 2—SWMB2[0:3] (0x5B18; RO) 1300
28-44 Software Mailbox 3—SWMB3[0:3] (0x5B1C; RO)....1300
28-45 EICR Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1301
28-46 EICR Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)....1302
28-47 EICS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0) 1302
28-48 SEICS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)....1303
28-49 EIMS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)....1303
28-50 EIMS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) 1304
28-51 EIMC Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)....1305
28-52 EIMC Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) 1305
28-53 Extended Interrupt Auto Clear—EIAC [0:3] (0x152C; R/W)....1306
28-54 IAM Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)....1307
28-55 EIAM Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1) 1307
28-56 Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C)....1308
28-57 Interrupt Cause Set Register—ICS [0:3] (0x1504; WO)....1310
28-58 Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W)....1312
28-59 Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO) 1314
28-60 Interrupt Acknowledge Auto Mask Register—IAM [0:3] (0x1510; R/W) 1316
28-61 Interrupt Throttle—EITR [0:3][0:9] (0x1680 + 4*n [n = 0...9]; R/W) 1317
28-62 Interrupt Vector Allocation Registers—IVAR [0:3][0:3](0x1700 + 4*n [n=0...3]; RW)..... 1318
28-63 Interrupt Vector Allocation Registers—MISC IVAR_MISC[0:3] (0x1740; RW) 1319
28-64 General Purpose Interrupt Enable—GPIE [0:3] (0x1514; RW)....1320
28-65 MSI-X Table Structure 1320
28-66 MSI-X PBA Structure 1321
28-67 MSIX Table Entry Lower Address—MSIXTADD [0:3][0:9] (BAR3: 0x0000 + 0x10*n [n=0...9]; R/W) 1321
28-68 MSIX Table Entry Upper Address—MSIXTUADD [0:3][0:9] (BAR3: 0x0004 + 0x10*n [n=0...9]; R/W) 1322

28-69 MSIX Table Entry Message—MSIXTMSG [0:3][0:9] (BAR3: 0x0008 + 0x10*n [n=0...9]; R/W)..... 1322
28-70 MSIX Table Entry Vector Control—MSIXTVCTRL [0:3][0:9] (BAR3: 0x000C + 0x10*n [n=0...9]; R/W) 1323
28-71 MSIXPBA Bit Description—MSIXPBA [0:3] (BAR3: 0x2000; RO)..... 1323
28-72 MSIX PBA Clear—PBACL [0:3] (0x5B68; R/W1C) 1324
28-73 Receive Control Register—RCTL[0:3] (0x0100; R/W)..... 1325
28-74 Split and Replication Receive Control—SRRCTL [0:3][0:7] (0xC00C + 0x40*n [n=0...7]; R/W)..... 1328
28-75 Split and Replication Receive Control—SRRCTL [0:3][1:7](0xC00C + 0x40*n [n=0...7]; R/W)..... 1329
28-76 Packet Split Receive Type—PSRTYPE [0:3][0:7] (0x5480 + 4*n [n=0...7]; R/W)..... 1331
28-77 Replicated Packet Split Receive Type—RPLPSRTYPE [0:3] (0x54C0; R/W)..... 1332
28-78 Receive Descriptor Base Address Low—RDBAL [0:3][0:7] (0xC000 + 0x40*n [n=0...7]; R/W)..... 1333
28-79 Receive Descriptor Base Address High—RDBAH [0:3][0:7] (0xC004 + 0x40*n [n=0...7]; R/W)..... 1333
28-80 Receive Descriptor Ring Length—RDLEN [0:3][0:7](0xC008 + 0x40*n [n=0...7]; R/W) .. 1334
28-81 Receive Descriptor Head—RDH [0:3][0:7] (0xC010 + 0x40*n [n=0...7]; RO)..... 1334
28-82 Receive Descriptor Tail—RDT [0:3][0:7] (0xC018 + 0x40*n [n=0...7]; R/W)..... 1335
28-83 Receive Descriptor Control—RXDCTL [0:3][0:7] (0xC028 + 0x40*n [n=0...7]; R/W)1336
28-84 Receive Queue Drop Packet Count—RQDPC [0:3][0:7] (0xC030 + 0x40*n [n=0...7]; RC/W)..... 1337
28-85 Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W) 1338
28-86 Receive Long Packet Maximum Length—RLPML [0:3] (0x5004; R/W)..... 1340
28-87 Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W) 1340
28-88 Multicast Table Array—MTA [0:3][0:127] (0x5200 + 4*n [n=0...127]; R/W) 1341
28-89 Receive Address Low 0—RAL0 [0:3][0:15] (0x5400 + 8*n [n=0...15]; 0x54E0 + 8*n [n=0...7]; R/W) 1342
28-90 Receive Address Low 1—RAL1 [0:3][0:7] (0x54E0 + 8*n [n=0...7]; R/W) 1343
28-91 Receive Address High 0—RAH0 [0:3][0:15] (0x5404 + 8*n [n=0...15]; 0x54E04 + 8*n [n=0...7]; R/W) 1344
28-92 Receive Address High 1—RAH1 [0:3][0:7] (0x54E4 + 8*n [n=0...7]; R/W) 1345
28-93 VLAN Filter Table Array—VFTA [0:3][0:127] (0x5600 + 4*n [n=0...127]; R/W)..... 1346
28-94 Multiple Receive Queues Command Register—MRQC [0:3](0x5818; R/W)..... 1347
28-95 RSS Random Key Register—RSSRK [0:3][0:9] (0x5C80 + 4*n [n=0...9]; R/W) 1349
28-96 Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W) 1349
28-97 Immediate Interrupt RX—IMIR [0:3][0:7] (0x5A80 + 4*n [n=0...7]; R/W) 1351
28-98 Immediate Interrupt Rx Ext.—IMIREXT [0:3][0:7] (0x5AA0 + 4*n [n=0...7]; R/W). 1352
28-99 2tuples Queue Filter—TTQF[0:3][0:7] (0x59E0 + 4*n[n=0..7]; R/W) 1353
28-100 Immediate Interrupt Rx VLAN Priority—IMIRVP [0:3] (0x5AC0; R/W) 1354
28-101 SYN Packet Queue Filter—SYNQF [0:3] (0x55FC; RW) 1354
28-102 EType Queue Filter—ETQF [0:3][0:7] (0x5CB0 + 4*n[n=0...7]; RW)..... 1355
28-103 Transmit Control Register—TCTL [0:3] (0x0400; R/W)..... 1356
28-104 Transmit Control Extended—TCTL_EXT [0:3] (0x0404; R/W)..... 1357
28-105 Transmit IPG Register—TIPG [0:3] (0x0410; R/W)..... 1358
28-106 Retry Buffer Control—RETX_CTL [0:3] (0x041C; R/W)..... 1359
28-107 DMA Tx Control—DTXCTL [0:3] (0x3590; R/W) 1359
28-108 DMA TX TCP Flags Control Low—DTXTCPFLGL [0:3] (0x359C; R/W) 1360
28-109 DMA TX TCP Flags Control High—DTXTCPFLGH [0:3] (0x35A0; RW) 1361
28-110 DMA TX Max Total Allow Size Requests—DTXMXSZRQ [0:3] (0x3540; RW)..... 1361
28-111 DMA TX Maximum Packet Size—DTXMXPKTSZ [0:3] (0x355C; RW) 1362
28-112 Transmit Descriptor Base Address Low—TDBAL [0:3][0:7] (0xE000 + 0x40*n [n=0...7]; R/W)..... 1362
28-113 Transmit Descriptor Base Address High—TDBAH [0:3][0:7] (0xE004 + 0x40*n [n=0...7]; R/W)..... 1363

28-114 Transmit Descriptor Ring Length—TDLEN [0:3][0:7] (0xE008 + 0x40*n [n=0...7]; R/W). 1363
28-115 Transmit Descriptor Head—TDH [0:3] [0:7] (0xE010 + 0x40*n [n=0...7]; RO) 1364
28-116 Transmit Descriptor Tail—TDT [0:3][0:7] (0xE018 + 0x40*n [n=0...7]; R/W) 1364
28-117 Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W)..... 1365
28-118 Tx Descriptor Completion Write-Back Address Low—TDWBAL [0:3][0:7] (0xE038 + 0x40*n [n=0...7]; R/W).....1367
28-119 Tx Descriptor Completion Write-Back Address High - TDWBAH [0:3][0:7] (0xE03C + 0x40*n [n=0...7];R/W).....1367
28-120 Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W) ...1368
28-121 Tx DCA Control Registers—TXCTL [0:3][0:7] (0xE014 + 0x40*n [n=0...7]; R/W) 1370
28-122 DCA Requester ID Information—DCA_ID [0:3] (0x5B70; RO).....1371
28-123 DCA Control—DCA_CTRL [0:3] (0x5B74; R/W).....1371
28-124 VMDq Control Register—VT_CTL [0:3] (0x581C; R/W) 1372
28-125 Malicious Driver Free Block—MDFB [0:3] (0x3558; RWS).....1373
28-126 Last VM Misbehavior Cause—LVMMC[0:3] (0x3548; RC).....1373
28-127 VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW) 1374
28-128 Replication Offload Register—RPLOLR [0:3] (0x5AF0; RW 1375
28-129 VLAN VM Filter—VLVF [0:3][0:31] (0x5D00 + 4*n [n=0...31]; RW).....1376
28-130 Unicast Table Array—UTA [0:3][0:127] (0xA000 + 4*n [n=0...127]; R/W) 1377
28-131 Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW) 1377
28-132 Storm Control Status—SCSTS [0:3] (0x5DB4;RO) 1378
28-133 Broadcast Storm Control Threshold—BSCTRH [0:3] (0x5DB8;RW) 1379
28-134 Multicast Storm Control Threshold—MSCTRH [0:3] (0x5DBC; RW) 1379
28-135 Broadcast Storm Control Current Count - BSCCNT [0:3] (0x5DC0;RO) 1379
28-136 Multicast Storm Control Current Count—MSCCNT [0:3] (0x5DC4;RO).....1380
28-137 Storm Control Time Counter—SCTC [0:3] (0x5DC8; RO) 1380
28-138 Storm Control Basic Interval—SCBI [0:3] (0x5DCC; RW).....1381
28-139 Virtual Mirror Rule Control—VMRCTL [0:3][0:3] (0x5D80 + 0x4*n [n= 0...3]; RW)..1381
28-140 Virtual Mirror Rule VLAN—VMRVLAN [0:3][0:3](0x5D90 + 0x4*n [n= 0...3]; RW) ...1382
28-141 Virtual Mirror Rule VM—VMRVM [0:3][0:3] (0x5DA0 + 0x4*n [n= 0...3]; RW) 1382
28-142 DMA Receive Power Saving Register—DMARPS [0:3] (0x2500; R/W).....1383
28-143 DMA Transmit Power Saving Register—DMATPS [0:3] (0x2504; R/W) 1384
28-144 Watchdog Setup—WDSTP [0:3] (0x1040; R/W).....1385
28-145 Watchdog Software Device Status—WDSWSTS [0:3] (0x1044; R/W).....1386
28-146 Free Running Timer—FRTIMER [0:3] (0x1048; RWS) 1386
28-147 TCP Timer—TCPTIMER [0:3] (0x104C; R/W).....1387
28-148 RX Time Sync Control Register—TSYNCRXCTL [0:3] (0xB620;RW).....1388
28-149 RX timestamp Low—RXSTMPL [0:3] (0xB624; RO) 1389
28-150 RX Timestamp High—RXSTMPH [0:3] (0xB628; RO) 1389
28-151 RX Timestamp Attributes Low—RXSATRL[0:3] (0xB62C; RO) 1389
28-152 RX timestamp Attributes High—RXSATRH [0:3] (0xB630; RO).....1390
28-153 TX Time Sync Control Register—TSYNCTXCTL [0:3] (0xB614; RW) 1390
28-154 TX Timestamp Value Low—TXSTMPL [0:3] (0xB618;RO) 1391
28-155 TX Timestamp Value High—TXSTMPH[0:3] (0xB61C; RO).....1391
28-156 System Time Register Residue—SYSTIMR [0:3] (0xB6F8; RW) 1391
28-157 System Time Register Low—SYSTIML [0:3] (0xB600; RW).....1392
28-158 System Time Register High—SYSTIMH [0:3] (0xB604; RW) 1392
28-159 Time Adjustment Offset Register Low—TIMADJL [0:3] (0xB60C; RW).....1393
28-160 Time Adjustment Offset Register High—TIMADJH [0:3] (0xB610;RW).....1393
28-161 TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW).....1394
28-162 Target Time Register 0 Low—TRGTTIML0 [0:3] (0xB644; RW).....1396
28-163 Target Time Register 0 High—TRGTTIMH0 [0:3] (0xB648; RW) 1396
28-164 Target Time Register 1 Low—TRGTTIML1 [0:3] (0xB64C; RW) 1396
28-165 Target Time Register 1 High—TRGTTIMH1 [0:3] (0xB650; RW) 1397
28-166 Frequency Out 0 Control Register—FREQOUT0 [0:3] (0xB654; RW) 1397
28-167 Frequency Out 1 Control Register—FREQOUT1 [0:3] (0xB658; RW) 1398
28-168 Auxiliary Time Stamp 0 Register Low—AUXSTMPL0 [0:3] (0xB65C; RO) 1398

28-169 Auxiliary Time Stamp 0 Register High—AUXSTMPH0 [0:3] (0xB660; RO) 1399
28-170 Auxiliary Time Stamp 1 Register Low—AUXSTMPL1 [0:3] (0xB664; RO)..... 1399
28-171 Auxiliary Time Stamp 1 Register High—AUXSTMPH1 [0:3] (0xB668; RO) 1400
28-172 Time Sync RX Configuration—TSYNCRXCFG [0:3] (0x5F50; R/W)..... 1400
28-173 Time Sync SDP Configuration Register—TSSDP [0:3] (0x003C; R/W)..... 1401
28-174 Time Sync Interrupt Cause Register—TSICR [0:3] (0xB66C; RC/W1C)..... 1402
28-175 Time Sync Interrupt Mask Register—TSIM [0:3] (0xB674; RW)..... 1403
28-176 Time Sync Interrupt Set Register—TSIS [0:3] (0xB670; WO) 1404
28-177 PCS Configuration—PCS_CFG [0:3] (0x4200; R/W) 1405
28-178 PCS Link Control—PCS_LCTL [0:3] (0x4208; RW) 1405
28-179 PCS Link Status—PCS_LSTS [0:3] (0x420C; RO) 1407
28-180 AN Advertisement—PCS_ANADV [0:3] (0x4218; R/W)..... 1409
28-181 Link Partner Ability—PCS_LPAB [0:3] (0x421C; RO) 1410
28-182 Next Page Transmit—PCS_NPTX [0:3] (0x4220; RW) 1411
28-183 Link Partner Ability Next Page—PCS_LPABNP [0:3] (0x4224; RO)..... 1412
28-184 SFP I2C Command - I2CCMD [0:3] (0x1028; R/W) 1413
28-185 SFP I2C Parameters—I2CPARAMS [0:3] (0x102C; R/W)..... 1414
28-186 CRC Error Count—CRCERRS [0:3] (0x4000; RC)..... 1415
28-187 Alignment Error Count—ALGNERRC [0:3] (0x4004; RC)..... 1416
28-188 Symbol Error Count—SYMERRS [0:3] (0x4008; RC) 1416
28-189 Missed Packets Count—MPC [0:3] (0x4010; RC) 1417
28-190 Single Collision Count—SCC [0:3] (0x4014; RC) 1417
28-191 Excessive Collisions Count—ECOL [0:3] (0x4018; RC) 1418
28-192 Multiple Collision Count—MCC [0:3] (0x401C; RC) 1418
28-193 Late Collisions Count—LATECOL [0:3] (0x4020; RC)..... 1419
28-194 Collision Count—COLC [0:3] (0x4028; RC)..... 1419
28-195 Defer Count—DC [0:3] (0x4030; RC) 1420
28-196 Host Transmit Discarded Packets by MAC Count—HTDPMC [0:3] (0x403C; RC)..... 1420
28-197 Receive Length Error Count—RLEC [0:3] (0x4040; RC)..... 1421
28-198 XON Received Count—XONRXC [0:3] (0x4048; RC)..... 1421
28-199 XON Transmitted Count—XONTXC [0:3] (0x404C; RC) 1422
28-200 XOFF Received Count—XOFFRXC [0:3] (0x4050; RC) 1422
28-201 XOFF Transmitted Count—XOFFTXC [0:3] (0x4054; RC) 1423
28-202 FC Received Unsupported Count—FCRUC [0:3] (0x4058; RC)..... 1423
28-203 Packets Received [64 Bytes] Count—PRC64 [0:3] (0x405C; RC) 1424
28-204 Packets Received [65-127 Bytes] Count—PRC127 [0:3] (0x4060; RC) 1424
28-205 Packets Received [128-255 Bytes] Count—PRC255 [0:3] (0x4064; RC) 1425
28-206 Packets Received [256-511 Bytes] Count—PRC511 [0:3] (0x4068; RC) 1426
28-207 Packets Received [512-1023 Bytes] Count—PRC1023 [0:3] (0x406C; RC)..... 1426
28-208 Packets Received [1024 to Max Bytes] Count—PRC1522 [0:3] (0x4070; RC) 1427
28-209 Good Packets Received Count—GPRC [0:3] (0x4074; RC) 1427
28-210 Broadcast Packets Received Count - BPRC [0:3] (0x4078; RC) 1428
28-211 Multicast Packets Received Count—MPRC [0:3] (0x407C; RC)..... 1428
28-212 Good Packets Transmitted Count—GPTC [0:3] (0x4080; RC) 1429
28-213 Good Octets Received Count—GORCL [0:3] (0x4088; RC)..... 1429
28-214 Good Octets Received Count—GORCH [0:3] (0x408C; RC) 1430
28-215 Good Octets Transmitted Count—GOTCL [0:3] (0x4090; RC)..... 1430
28-216 Good Octets Transmitted Count—GOTCH [0:3] (4094; RC)..... 1431
28-217 Receive No Buffers Count—RNBC [0:3] (0x40A0; RC)..... 1431
28-218 Receive Undersize Count—RUC [0:3] (0x40A4; RC)..... 1432
28-219 Receive Fragment Count—RFC [0:3] (0x40A8; RC) 1432
28-220 Receive Oversize Count—ROC [0:3] (0x40AC; RC) 1433
28-221 Receive Jabber Count—RJC [0:3] (0x40B0; RC)..... 1433
28-222 Management Packets Received Count—MNGPRC [0:3] (0x40B4; RC) 1434
28-223 Management Packets Dropped Count—MPDC [0:3] (0x40B8; RC)..... 1434
28-224 Management Packets Transmitted Count—MNGPTC [0:3] (0x40BC; RC)..... 1435
28-225 Total Octets Received—TORL [0:3] (0x40C0; RC) 1435
28-226 Total Octets Received—TORH [0:3] (0x40C4; RC)..... 1436
28-227 Total Octets Transmitted—TOTL [0:3] (0x40C8; RC) 1436

28-228 Total Octets Transmitted—TOTH [0:3] (0x40CC; RC) 1437
28-229 Total Packets Received—TPR [0:3] (0x40D0; RC) 1437
28-230 Total Packets Transmitted—TPT [0:3] (0x40D4; RC 1438
28-231 Packets Transmitted [64 Bytes] Count—PTC64 [0:3] (0x40D8; RC) 1438
28-232 Packets Transmitted [65-127 Bytes] Count—PTC127 [0:3] (0x40DC; RC) 1439
28-233 Packets Transmitted [128-255 Bytes] Count—PTC255 [0:3] (0x40E0; RC) 1439
28-234 Packets Transmitted [256-511 Bytes] Count—PTC511 [0:3] (0x40E4; RC) 1440
28-235 Packets Transmitted [512-1023 Bytes] Count—PTC1023 [0:3] (0x40E8; RC) 1440
28-236 Packets Transmitted [1024 Bytes or Greater] Count—PTC1522 [0:3] (0x40EC; RC) .1441
28-237 Multicast Packets Transmitted Count—MPTC [0:3] (0x40F0; RC)....1441
28-238 Broadcast Packets Transmitted Count—BPTC [0:3] (0x40F4; RC)....1442
28-239 TCP Segmentation Context Transmitted Count—TSCTC [0:3] (0x40F8; RC)....1442
28-240 Interrupt Assertion Count—IAC [0:3] (0x4100; RC) 1443
28-241 Rx Packets to Host Count—RPTHC [0:3] (0x4104; RC) 1443
28-242 Debug Counter 1—DBGC1 [0:3] (0x4108; RC) 1443
28-243 DBGC1 Values 1444
28-244 Debug Counter 2—DBGC2 [0:3] (0x410C; RC) 1444
28-245 DBGC2 Values 1444
28-246 Debug Counter 3—DBGC3 [0:3] (0x4110; RC) 1445
28-247 DBGC3 Values 1445
28-248 Debug Counter 4—DBGC4 [0:3] (0x411C; RC) 1445
28-249 DBGC4 Values 1446
28-250 Host Good Packets Transmitted Count—HGPTC [0:3] (0x4118; RC) 1446
28-251 Receive Descriptor Minimum Threshold Count—RXDMTC [0:3] (0x4120; RC)....1446
28-252 Host Good Octets Received Count—HGORCL [0:3] (0x4128; RC) 1447
28-253 Host Good Octets Received Count—HGORCH [0:3] (0x412C; RC) 1447
28-254 Host Good Octets Transmitted Count—HGOTCL [0:3] (0x4130; RC) 1448
28-255 Host Good Octets Transmitted Count - HGOTCH [0:3] (0x4134; RC) 1448
28-256 Length Error Count—LENERRS [0:3] (0x4138; RC) 1449
28-257 SerDes/SGMII/KX Code Violation Packet Count—SCVPC [0:3] (0x4228; RW)....1449
28-258 Switch Drop Packet Count—SDPC [0:3] (0x41A4; RC)....1450
28-259 Per Queue Good Packets Received Count—VFGPRC [0:3][0:7](0x10010 + n*0x100 [n=0...7]; RO)....1450
28-260 Per Queue Good Packets Transmitted Count—VFGPTC [0:3][0:7](0x10014 + n*0x100 [n=0...7]; RO)....1451
28-261 Per Queue Good Octets Received Count—VFGORC [0:3][0:7] (0x10018 + n*0x100 [n=0...7]; RO)....1452
28-262 Per Queue Good Octets Transmitted Count—VFGOTC [0:3][0:7] (0x10034 + n*0x100 [n=0...7]; RO)....1453
28-263 Per Queue Multicast Packets Received Count—VFMPRC [0:3][0:7] (0x10038 + n*0x100 [n=0...7]; RO)....1453
28-264 BMC Management Packets Dropped Count—BMPDC [0:3] (0x4140; RC) 1454
28-265 BMC Management Packets Transmitted Count—BMNGPTC [0:3] (0x4144; RC)....1454
28-266 BMC Management Packets Received Count—BMNGPRC [0:3] (0x413C; RC)....1455
28-267 BMC Total Unicast Packets Received—BUPRC [0:3] (0x4400; RC) 1455
28-268 BMC Total Multicast Packets Received—BMPRC [0:3] (0x4404; RC)....1456
28-269 BMC Total Broadcast Packets Received—BBPRC [0:3] (0x4408; RC)....1456
28-270 BMC Total Unicast Packets Transmitted—BUPTC [0:3] (0x440C; RC) 1456
28-271 BMC Total Multicast Packets Transmitted—BMPTC [0:3] (0x4410; RC)....1457
28-272 BMC Total Broadcast Packets Transmitted—BBPTC [0:3] (0x4414; RC)....1457
28-273 BMC FCS Receive Errors—BCRCERRS [0:3] (0x4418; RC) 1458
28-274 BMC Alignment Errors—BALGNERRC [0:3] (0x441C; RC) 1458
28-275 BMC Pause XON Frames Received—BXONRXC [0:3] (0x4420; RC) 1458
28-276 BMC Pause XOFF Frames Received—BXOFFRXC [0:3] (0x4424; RC)....1459
28-277 BMC Pause XON Frames Transmitted—BXONTXC [0:3] (0x4428; RC) 1459
28-278 BMC Pause XOFF Frames Transmitted—BXOFFTXC [0:3] (0x442C; RC)....1460
28-279 BMC Single Collision Transmit Frames—BSCC [0:3] (0x4430; RC)....1460
28-280 BMC Multiple Collision Transmit Frames—BMCC [0:3] (0x4434; RC) 1460
28-281 Wakeup Control Register—WUC [0:3] (0x5800; R/W) 1461

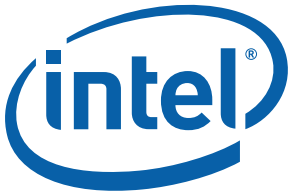
28-282 Wakeup Filter Control Register—WUFC [0:3] (0x5808; R/W) 1462
28-283 Wakeup Status Register—WUS [0:3] (0x5810; R/W1C) 1463
28-284 Wakeup Packet Length—WUPL [0:3] (0x5900; RO) 1464
28-285 Wakeup Packet Memory—WUPM [0:3][0:31] (0x5A00 + 4*n [n=0...31]; RO)..... 1464
28-286 IP Address Valid—IPAV [0:3] (0x5838; R/W)..... 1465
28-287 IPv4 Address Table—IP4AT [0:3][0:3] (0x5840 + 8*n [n=0...3]; RW)..... 1465
28-288 IPv6 Address Table—IP6AT [0:3][0:3] (0x5880 + 4*n [n=0...3]; RW)..... 1466
28-289 Flex Filter Even Data Register Fields—FEDR [0:3][0:3][0:15] (0x9000 +16*n[0..15]; RW) 1467
28-290 Flex Filter Odd Data Register Fields—FODR [0:3][0:3][0:15] (0x9000 +16*n[0..15];RW) 1468
28-291 Flex Filter Mask Field Register—FMFR [0:3][0:3][0:15] (0x9008 +16*n[0..15];RW) 1468
28-292 Flex Filter Queueing Field—FQFR [0:3][0:3][0:15] (0x90FC + 16*n[n=0..15] ;RW). 1469
28-293 Flex Filter Even Data Register Extended—FHFT_EXT_FEDR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)..... 1470
28-294 Flex Filter Odd Data Register Extended—FHFT_EXT_FODR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)..... 1471
28-295 Flex Filter Mask Field Extended—FHFT_EXT_FMFR [0:3][0:3][0:15] (0x9A08 +16*n[0..15]; RW)..... 1471
28-296 Flex Filter Queueing Extended—FHFT_EXT_FQFR [0:3][0:3] (0x9AFC; RW) 1472
28-297 Management VLAN TAG Value—MAVTV [0:3] [0:7] (0x5010 +4*n [n=0...7]; RW) .. 1473
28-298 Management Flex UDP/TCP Ports—MFUTP [0:3][0:3] (0x5030 + 4*n [n=0...3]; RW)..... 1473
28-299 Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW)..... 1474
28-300 Management Control Register—MANC [0:3] (0x5820; RW)..... 1475
28-301 Management Only Traffic Register—MNGONLY [0:3] (0x5864; RW) 1476
28-302 Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW)..... 1476
28-303 Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW)..... 1478
28-304 Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW)..... 1481
28-305 Manageability MAC Address Low—MMAL [0:3][0:1] (0x5910 + 8*n [n= 0...1]; RW) 1481
28-306 Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0...1]; RW) 1482
28-307 Parity and ECC Error Indication—PEIND [0:3] (0x1084; RC) 1484
28-308 Parity and ECC Indication Mask—PEINDM [0:3] (0x1088; RW)..... 1485
28-309 DMA Transmit Descriptor Parity Status—DTPARS [0:3] (0x3510; RW1C)..... 1485
28-310 DMA Receive Descriptor Parity Status—DRPARS [0:3] (0x3514; RW1C) 1486
28-311 Dhost Parity Status—DDPARS [0:3] (0x3518; RW1C) 1486
28-312 Tx Packet Buffer ECC Status—TPBECCSTS [0:3] (0x345C; RW)..... 1487
28-313 LAN Port Parity Error Control Register—LANPERRCTL [0:3] (0x5F54; RW) 1487
28-314 LAN Port Parity Error Status Register—LANPERRSTS [0:3] (0x5F58; RO)..... 1488
29-1 Bus B, Device D, Function 1: Summary of PCIe* GigE Configuration Registers 1492
29-2 Bus B, Device D, Function 2: Summary of PCIe* GigE Configuration Registers 1492
29-3 Bus B, Device D, Function 3: Summary of PCIe* GigE Configuration Registers 1492
29-4 Bus B, Device D, Function 4: Summary of PCIe* GigE Configuration Registers 1492
29-5 Bus B, Device D, Function 1+Index1: Summary of PCIe* GigE Configuration Registers ... 1492
29-6 Bus B, Device D, Function 1+Index 1: Summary of GigE Registers Mapped Through GbEPCI Memory BAR..... 1494
29-7 PVID[0:3]—PF Vendor Identification Register 1495
29-8 PDID0—PF Device Identification Register (GbE0)..... 1496
29-9 PDID1—PF Device Identification Register (GbE1)..... 1496
29-10 PDID2—PF Device Identification Register (GbE2)..... 1497
29-11 PDID3—PF Device Identification Register (GbE3)..... 1497
29-12 PPCICMD[0:3]—PF Device Command Register 1498
29-13 PPCISTS[0:3]—PF PCI Device Status Register..... 1499
29-14 PRID[0:3]—PF Revision ID Register..... 1501
29-15 PCC[0:3]—PF Class Code Register..... 1501

29-16 PHDR[0:3]—PF Header Type Register 1502
29-17 GbE Controller Base Address Registers Description - LAN 0...3 1502
29-18 Base Address Registers' Fields 1502
29-19 GbEPCIBAR0[0:3]—BAR0 Base Address Register 1503
29-20 GbEPCIBAR1[0:3]—BAR1 Base Address Register 1504
29-21 GbEPCIBAR2[0:3]—BAR2 Base Address Register 1504
29-22 GbEPCIBAR3[0:3]—BAR3 Base Address Register 1505
29-23 GbEPCIBAR4[0:3]—BAR4 Base Address Register 1506
29-24 GbEPCIBAR5[0:3]—BAR5 Base Address Register 1507
29-25 PSVID[0:3]—PF Subsystem Vendor ID Register 1507
29-26 PSID[0:3]—PF Subsystem ID Register 1508
29-27 PCP[0:3]—PF Capabilities Pointer Register 1508
29-28 PIRQL[0:3]—PF Interrupt Line Register 1509
29-29 PIRQP0—PF Interrupt Pin Register 0 1509
29-30 PIRQP1—PF Interrupt Pin Register 1 1510
29-31 PIRQP2—PF Interrupt Pin Register 2 1510
29-32 PIRQP3—PF Interrupt Pin Register 3 1511
29-33 PPMCAP[0:3]—PF Power Management Capabilities ID Register 1512
29-34 PPMCP[0:3]—PF Power Management Next Capability Pointer Register 1512
29-35 PPMC[0:3]—PF Power Management Capabilities Register 1513
29-36 PPMCSR[0:3]—PF Power Management Control and Status Register 1514
29-37 PMSICID[0:3] - Message Signalled Interrupt Capability ID Register 1515
29-38 PMSINCP[0:3]—Message Signalled Interrupt Next Capability Pointer Register 1515
29-39 PMSICTL[0:3]—Message Signalled Interrupt Control Register 1516
29-40 PMSILADDR[0:3]—Message Signalled Interrupt Lower Address Register 1516
29-41 PMSIUADDR[0:3]—Message Signalled Interrupt Upper Address Register 1517
29-42 PMSIDATA[0:3]—Message Signalled Interrupt Data Register 1517
29-43 PMSIMSK[0:3]—Message Signalled Interrupt Mask Register 1517
29-44 PMSIPND[0:3]—Message Signalled Interrupt Pending Register 1518
29-45 MSI-X Capability Structure 1519
29-46 PMSI-X[0:3]—PF Message Signalled Interrupt X Capability ID Register 1519
29-47 PMSIXNCP[0:3]—PF MSIX Next Capability Pointer Register 1520
29-48 PMSIXCNTL[0:3]—PF Message Signalled Interrupt X Control Register 1520
29-49 PMSIXTBIR[0:3]—PF MSI-X Table Offset & Table BIR Register 1521
29-50 PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register 1521
29-51 IOADDR[0:3]—IOADDR Register 1522
29-52 IODATA[0:3]—IODATA Register 1522
29-53 PPCID[0:3]—PF PCI Express Capability Register 1524
29-54 PPCP[0:3]—PF PCI Express Next Capability Pointer Register 1524
29-55 PPCR[0:3]—PF PCI Express Capabilities Register 1524
29-56 PPDCAP[0:3]—PF PCI Express Device Capabilities Register 1525
29-57 PPDCNTL[0:3]—PF PCI Express Device Control Register 1527
29-58 PPDSTAT[0:3]—PF PCI Express Device Status Register 1528
29-59 PLCAPR[0:3]—PF Link Capabilities Register 1529
29-60 PLCNTLR[0:3]—PF Link Control Register 1531
29-61 PLSR[0:3]—PF Link Status Register 1532
29-62 PDCAPR2[0:3]—PF Device Capabilities 2 Register 1534
29-63 PDCNTR2[0:3]—PF Device Control 2 Register 1535
29-64 PLCNTLR2[0:3]—PF Link Control 2 Register 1536
29-65 PLSR2[0:3]—PF Link Status 2 Register 1538
29-66 VPDCID[0:3]—VPD Capability ID Register 1539
29-67 VPDNCP[0:3]—VPD Next Capability Pointer Register 1539
29-68 VPDADDR[0:3]—VPD Address Register 1540
29-69 VPDDATA[0:3]—VPD Data 1540
29-70 PCIe* Extended Configuration Space 1541
29-71 PCIe* Extended Capability Structure 1541
29-72 PPCIEAERCAPID[0:3]—PF PCI Express AER Capability ID Register 1542
29-73 PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register 1542
29-74 PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register 1543

29-75 PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register 1544
29-76 PPAERCS[0:3]—PF PCI Express AER Correctable Error Register 1545
29-77 PPAERCM[0:3]—PF PCI Express AER Correctable Error Mask Register 1546
29-78 PPAERCTLCAP[0:3]—PF PCI Express AER Control and Capability Register 1546
29-79 PPAERHDRLOG0[0:3]—PF PCI Express AER Header Log 0 Register 1547
29-80 PPAERHDRLOG1[0:3]—PF PCI Express AER Header Log 1 Register 1547
29-81 PPAERHDRLOG2[0:3]—PF PCI Express AER Header Log 2 Register 1548
29-82 PPAERHDRLOG3[0:3]—PF PCI Express AER Header Log 3 Register 1548
29-83 PARIDHDR[0:3]—PF Alternative Routing ID Capability Header 1549
29-84 PFARICAP0—PF ARI Capabilities Register 1550
29-85 PFARICAP1—PF ARI Capabilities Register 1551
29-86 PFARICAP2—PF ARI Capabilities Register 1552
29-87 PFARICAP3—PF ARI Capabilities Register 1552
29-88 PARIDCTL[0:3]—PF Alternative Routing ID Control Register 1553
29-89 IOBAR Register Map 1554
30-1 PCH Die-level Legacy I/O (DMI)TAP Pin Interface 1557
30-2 PCH Die-Level Endpoint TAP Pin Interface 1558
31-1 Platform External Clock Interface 1562
32-1 Signal Type Definitions 1563
32-2 Direct Media Interface Signals 1566
32-3 PCI Express* Endpoint Interface Signals 1567
32-4 SerDes/SGMII Interface Signals 1570
32-5 SFP Interface Signals 1573
32-6 GbE EEPROM SPI Interface Signals 1576
32-7 GbE SMBus Interface Signals (Master/Slave) 1577
32-8 LED, Software Defined, Miscellaneous Signals 1578
32-9 EndPoint Management SMBus Interface Signals (Slave) 1585
32-10 PCI Express* Root Complex Interface Signals 1586
32-11 SATA Interface Signals 1587
32-12 LPC Interface Signals 1589
32-13 USB* Interface Signals 1590
32-14 UART Interface Signals 1591
32-15 Host SMBus (Master) Interface Signals 1594
32-16 SPI Boot Interface Signals 1596
32-17 Interrupt Interface Signals 1596
32-18 Processor Interface Signals 1597
32-19 Power Management Interface Signals 1598
32-20 Thermal Sensor Current Reference 1601
32-21 Miscellaneous Interface Signals 1601
32-22 General Purpose I/O Interface Signals 1602
32-23 Real Time Clock (RTC) Interface Signals 1611
32-24 Clock Input Interface Signals 1613
32-25 PCH JTAG Interface Signals 1615
32-26 EndPoint JTAG Interface Signals 1615
32-27 Strap Signals 1617
32-28 Reserved Signals 1624
32-29 Power and Ground Signals 1629
33-1 PCH Absolute Maximum and Minimum Ratings 1634
33-2 Operating Conditions Power Supply Rail 1635
33-3 Maximum Supply Current 1636
33-4 Power for Standby Support Designs 1637
33-5 Power for Non-Standby Support Designs 1638
33-6 DC Input Characteristic Signal Association 1640
33-7 DC Input Characteristics 1641
33-8 DC Output Characteristic Signal Association 1642
33-9 DC Output Characteristics 1643
33-10 PCI Express* and DMI Gen1 TX Specification Interface Timings 1644
33-11 PCI Express* and DMI Gen1 RX Specification Interface Timings 1645
33-12 PCI Express* Gen2 TX Specification Interface Timings 1647

33-13 PCI Express* Gen2 RX Specification Interface Timings 1648
33-14 SATA Specification Interface Timings 1650
33-15 GbE SGMII Driver DC Specification Interface....1651
33-16 GbE SGMII Receiver DC Specification Interface (AC Coupled) 1651
33-17 GbE SGMII Driver AC Specification Interface 1652
33-18 SerDes GbE 1000 Base RX Specifications....1652
33-19 SerDes GbE 1000 Base TX Specifications....1652
33-20 CRU Differential Input Clock DC Specifications....1653
33-21 CRU Differential Input Clock Timing Specifications 1655
33-22 SATA, DMI, and PCIe* Differential Input Clock DC Specification....1655
33-23 SATA, DMI, and PCIe* Differential Input Clock AC Specification 1656
33-24 Digital 3.3V I/O Timing Characteristics....1659
33-25 Clock Timings 1660
33-26 Universal Serial Bus (USB) Timing 1663
33-27 SMBus Timing....1663
33-28 LPC Timing....1664
33-29 SPI Timings (20 MHz) 1664
33-30 SPI Timings (33 MHz) 1664
33-31 SPI Timings (50 MHz) 1665
33-32 I^2C Timing Parameters....1666
33-33 DC Input Characteristics: MDIO Mode of Operation 1667
33-34 DC Output Characteristics: MDIO Mode of Operation 1667
33-35 MDIO I/F Timing Parameters....1668
33-36 EEPROM I/F Timing Parameters....1669
33-37 PECI Timings 1670
33-38 UART Received Timing 1671
33-39 UART Transmit Timing 1671
33-40 JTAG I/F Timing Parameters 1671
35-1 Alphabetical Ball Listing....1680
35-2 Alphabetical Signal Listing 1688
35-3 Ball Map Color Code....1696
35-4 Bottom View Left....1696
35-5 Bottom View Right....1699
35-6 Top View Left 1703
35-7 Top View Right....1706

§ §



Overview and PCH Interfaces - Volume 1 of 4

April 2014

Order Number: 327879-005US

§ §

1.0 Introduction

1.1 Introduction

The Intel® Communications Chipset 89xx Series can be most easily described as a PCH that includes standard PC interfaces (for example: PCI Express*, SATA*, USB*, and so on.) along with an EndPoint Interface which includes Intel® QuickAssist Technology and GbE MACs.

This document is intended as a reference for architects, hardware/software designers and engineers, or others who need technical information for silicon development and for programming the hardware. It is expected that the readers of this document have an understanding of:

- IA-64 microprocessor
- Memory controller
- I/O architecture (PCI Express)
- Intel® QuickAssist Technology
- A basic understanding of system software (operating system and BIOS) internals

In this document:

- *Chipset*, *Platform Controller Hub*, and *PCH* are used as generic references to the Intel® Communications Chipset 89xx Series.
- Intel® Communications Chipset 89xx Series.
  - DH89xxCC includes Chipset 89xx Series SKUs where 8900 ≤ *SKU* ≤ 8920
  - DH89xxCL includes Chipset 89xx Series SKUs where 8925 ≤ *SKU* ≤ 8955

1.2 Intel® Communications Chipset 89xx Series SKU Definition

Table 1-1 provides a high-level summary of the Chipset 89xx Series PCH SKUs.

- Intel® Communications Chipset 89xx Series.
  - — DH89xxCC includes Chipset 89xx Series SKUs where 8900 ≤ *SKU* ≤ 8920
  - — DH89xxCL includes Chipset 89xx Series SKUs where 8925 ≤ *SKU* ≤ 8955

Table 1-1. Intel® Communications Chipset 89xx Series SKUs

Devices	DH89xxCC				DH89xxCL			
	DH8900CC	DH8903CC	DH8910CC	DH8920CC	DH8925CL	DH8926CL	DH8950CL	DH8955CL
PCIe EndPoint (max width)	x4		x8	x16				
Virtual Function (SR-IOV) for Intel®QuickAssist Technology Services	0	16			32			
GbE	4				0			
SATA	2							
USB	6							
Normal Mode TDP (W)	8.5	9.5	11	12	17		20	
End Point Mode TDP (W)	5.7	6.7	8.2	9.2	14.2		17.2	
Non-End Point Mode TDP (W)	2.8							
$T_{JUNCTION-MAX}$ (°C)	103							
$T_{JUNCTION-MIN}$ (°C)	0							

Notes:
1. Contact your local Intel Field Sales Representative for currently available PCH SKUs.
2. Table 1-1 shows feature differences between the PCH SKUs. If a feature is not listed in Table 1-1, it is considered as a base feature that is included in all SKUs.

1.3 Differences Between DH89xxCC and DH89xxCL SKUs

This document merges the design requirements for all existing Intel® Communications Chipset 89xx Series. However, since the DH89xxCL SKU may be used in existing DH89xxCC SKU based designs, Table 1-2 points out the chapters, sections, tables and figures in the document where requirements differ between the two PCHs.

Table 1-2. Summary of Differences between DH89xxCC and DH89xxCL SKUs

Chapter	Section	Figure	Table	Comment
Chapter 2.0, "Architecture Overview"	Section 2.1	Figure 2-1		PCH Modes of Operation Block Diagram
	Section 2.2	Figure 2-2		PCH Architecture Block Diagram
Chapter 3.0, "PCH Platform Memory and Device Configuration"	Section 3.6.2		Table 3-3	PCH PCI Device Summary
			Table 3-4	EndPoint (EP) PCI Device Summary
Chapter 19.0, "PCI Express* EndPoint Introduction"	Section 19.1.3	Figure 19-1		PCIe* EP Interface Block Diagram
	Section 19.1.4	Figure 19-2		EP Functional Description Block Diagram
Chapter 20.0, "PCIe Endpoint Function 0 Registers"	Section 20.2.2.18		Table 20-23	PCH SKU Register
Chapter 32.0, "Signal Descriptions"	Section 32.2	Figure 32-1		PCH Interface Signals Block Diagram (Refer to ***Note 1*** at the bottom of Figure 32-1)
	Section 32.6		Table 32-4	SerDes/SGMII Interface Signals
			Table 32-5	SFP Interface Signals
			Table 32-6	GbE EEPROM Interface Signals
			Table 32-7	GbE SMBus Interface Signals
			Table 32-8	LED, Software Defined, Misc. Signals
	Section 32-22		Table 32-24	Clock Input Interface Signals
	Section 32-24		Table 32-27	Strapping Signals
	Section 32.25		Table 32-28	Reserved Signals
	Section 32.26		Table 32-29	Power and Ground Signals
Chapter 33.0, "Electrical Characteristics"	Section 33.3		Table 33-3	Maximum ICC Supply Current

1.4 Document Organization

This document begins with a description of the Intel® Communications Chipset 89xx Series PCH product line architecture, building blocks and usage models for the product line as well as their high-level programming model and memory map. The overview chapter also reviews the block diagram and defines external and internal interfaces. The document is organized into four volumes:

- Volume 1 – Overview and PCH Interfaces
- Volume 2 – PCIe Endpoint and Gigabit Ethernet
- Volume 3 – Test Features
- Volume 4 – Technical Specifications

1.5 Referenced Documents and Related Websites

Table 1-3. Referenced Documents

Document Title	Location
Advanced Configuration and Power Interface (ACPI) Specification	http://www.acpi.info/
Intel Corporation, Enhanced Host Controller Interface Specification for Universal Serial Bus	http://www.intel.com/technology/usb/ehcispec.htm
Intel Corporation, High Precision Event Timers (HPET) Specification	http://www.intel.com/hardwaredesign/hpetspec.htm
Intel Corporation, Low Pin Count (LPC) Interface Specification	http://www.intel.com/design/chipsets/industry/lpc.htm
Intel Corporation, Enhanced Host Controller Interface (EHCI) Specification	http://www.intel.com/technology/usb/ehcispec.htm
Intel Corporation, Universal Serial Bus (USB) Specification	http://www.intel.com/technology/usb/spec.htm
Intel Corporation, USB2 Debug Device Functional Specification	http://www.intel.com/technology/usb/download/DebugDeviceSpec_R090.pdf
JEDEC Specification	http://www.jedec.org/default.cfm
Serial ATA Specification	https://www.sata-io.org/developers/purchase_spec.asp
SMBus Specification	http://www.smbus.org/specs/
Universal Serial Bus Specification 1.1 and 2.0	http://www.usb.org/developers/docs/

Table 1-4. Related Websites

Specification or Technology	Website
PIRQ routing table information	http://www.microsoft.com/whdc/archive/pciirq.mspx
ACPI and related specifications	http://www.acpi.info/spec.htm
AT Attachment-6 with Packet Interface (ATA/ATAPI-6)	http://T13.org (T13 1410D)
BIOS boot specifications	http://www.phoenix.com/en/customer+services/white+papers-specs/
PCI and PCI Express* related specifications	http://www.pcisig.com/specifications
Power management specifications	http://www.microsoft.com/whdc/resources/respec/specs/pmref/default.mspx

1.6 Acronyms

This section describes acronyms that are used throughout this document.

Table 1-5. Acronym Table (Sheet 1 of 3)

Term	Description
ACPI	Advanced Configuration and Power Interface Specification, an industry specification of the common interfaces enabling robust operating system (OS)-directed motherboard device configuration and power management of both devices and entire systems.
AHCI	Advanced Host Controller Interface, an industry specification of the interface between memory and SATA devices.
ARP	Address resolution protocol
ASF	Alert Specification Format. This is the next generation of "Alert on LAN*" implementation.
BAR	PCI Base Address Register used to define the base and limit of an I/O or memory region assigned to a PCI device.
BER	Bit Error Rate
BGA	Ball Grid Array
CMC	Common Mode Choke
CM	Coherent Memory
CNR	Communications and Networking Riser
CRC	See Cyclic Redundancy Check in Table 1-6.
DDP	Direct Data Placement Protocol
DDR	DDR SDRAM (Double Data Rate Synchronous Dynamic Random Access Memory) is a system memory technology.
DED	Double-bit Error Detect
DMA	See Direct Memory Access in Table 1-6.
DW	Double Word. A legacy reference to 32 bits of data on a naturally aligned four-byte boundary (for example, the least significant two bits of the byte address are b00). This is a legacy term used by PCI and must not be used other than in that context.
ECC	Error Checking and Correction
EMI	Electro Magnetic Interference
EMTS	Electrical Mechanical Thermal Specification used for processor specifications.
ESD	Electrostatic Discharge
FRU	Field Replaceable Unit
FS	Full-speed. Refers to USB.
GbE	Gigabit Ethernet
GigE	Gigabit Ethernet
HBA	Host Bus Adapter - necessary when connecting a peripheral to a computer that doesn't have native support for that peripheral's interface.
HCD	Host Controller Device - USB interface for programmers
HECBASE	PCI Express Enhanced Configuration Base Register
HPET	High Precision Event Time (HPET) - The IA-PC HPET Architecture defines a set of timers that can be used by the operating system. The timers are defined such that the OS may be able to assign specific timers to be used directly by specific applications. Each timer can be configured to generate a separate interrupt.
HSI	High Speed Interface. Refers to USB.
IA	Intel Architecture instruction set commonly known as "x86"
IA-CPU	IA-CPU, IA Complex and IA Processor are the same terminology

Table 1-5. Acronym Table (Sheet 2 of 3)

Term	Description
INTx	Legacy PCI interrupt architecture that encodes interrupts on one of four side-band signals (INTA, INTB, INTC, and INTD).
I/O	• Input/Output. • When used as a qualifier to a transaction type, specifies that transaction targets Intel® architecture-specific I/O space (for example, I/O read).
IP	Internet Protocol
ISA	See Industry Standard Architecture in Table 1-6
LML	Latency Measurement Logic
LPC	Low Pin Count
LS	Low-speed. Refers to USB.
LSb	Least Significant Bit
LSB	Least Significant Byte
ME	Management Engine
MMIO	Memory Mapped I/O
MSb	Most Significant Bit
MSB	Most Significant Byte
MSI	Message-signaled interrupt that encodes interrupts as an in-band 32-bit write transaction.
MTBF	Mean Time Between Failures
NCM	Non Coherent Memory
NIC	Network interface controller
OS	Operating system.
OSPM	Operating system directed Power Management
P2P	See Peer-to-Peer in Table 1-6
PCH	Platform Controller Hub
PCI	Peripheral Component Interconnect Local Bus. A 32- or 64-bit bus with multiplexed address and data lines that is primarily intended for use as an interconnect mechanism within a system between processor/memory and peripheral components or add-in cards.
PCM	Pulse Code Modulation
PDP Platform	PECI-to-DIMM Processor (PDP) based platform: Platforms based on CPUs with integrated SMBuses to DIMMs. The PCH collects DIMM Thermal Data via the PECI interface to the Processor
PEC	Packet Error Checking. This is an SMBUS 2.0 feature.
POC	Power-on-configuration
QAT	Intel® QuickAssist Technology
RASUM	Reliability, Availability, Serviceability, Usability, and Manageability, which are important characteristics of servers.
RCBA	Root Complex Base Address.
RCRB	Root Complex Register Block, as defined in the *PCI Express Specification v1.0a*.
RDMA	Remote Direct Memory Access
RFL	Receive FIFO Level
RMW	Read-Modify-Write operation
RTC	Real-Time Clock
RTCRESET#	Signal that resets the RTC well (but does not clear the RTC RAM memory contents).
SATA	Serial Advanced Technology Attachment

Table 1-5. Acronym Table (Sheet 3 of 3)

Term	Description
SATA*	Serial ATA, an industry specification of the interface for storage controllers and devices.
SDP Platform	SMBus-to-DIMM Processor (PDP) based platform: Platforms based on CPUs without integrated SMBuses to DIMMs. The PCH collects DIMM Thermal Data via the platform internal Host SMBus interface to the DIMMs
SEC	Single-bit Error Correct
SEC/DED	Single Error Correct/Double Error Detect - A specific data protection algorithm that distributes data and ECC across 144 bits. Enables correction of single bit errors. Allows detection of double bit errors.
SMM	System Management Mode
SPD	Serial Presence Detect
STR	Suspend To Ram
TAP	Test Access Port used for testability and debug of the component.
TCO	Total Cost of Ownership
TCP	Transmission Control Protocol
TDR	Time Domain Reflectometry
TFL	Transit FIFO Level
TID	See Transaction Identifier in Table 1-6
USB	Universal Serial Bus
VSCC	Vendor Specific Component Capabilities
WDT	Watch Dog Timer
XX	Classification used to describe a register's reset value when the value of the register is indeterminate. Certain registers' value at reset does not have a default value, therefore it is unknown what the value will be.

1.7 Glossary

This section presents a glossary for this document.

Table 1-6. Glossary Table (Sheet 1 of 5)

Term	Definition
Agent	A logical device connected to a bus or shared interconnect that can either initiate accesses or be the target of accesses.
ALT Access Mode	Mode to allow the reading of write-only registers, usually used when saving/restoring register content for power management sleep state implementations.
Anti-Etch	Any plane-split, void or cutout in a V_{CC} or GND plane is referred to as an anti-etch.
Asserted	Signal is set to a level that represents logical true.
Asynchronous	• An event that causes a change in state with no relationship to a clock signal. • When applied to transactions or a stream of transactions, a classification for those that do not require service within a fixed time interval.
Atomic operation	A series of two or more transactions to a device by the same initiator which are guaranteed to complete without intervening accesses by a different master. Most commonly required for a read-modify-write (RMW) operation.
Buffer	• A random access memory structure. • The term I/O buffer is also used to describe a low-level input receiver and output driver combination.
Cx States	Processor power states (Cx states) are processor power consumption and thermal management states within the global working state, G0. • C0: Processor power state - While the processor is in this state, it executes instructions. • C1: Processor power state - This power state has the lowest latency. The hardware latency in this state must be low enough that the operating software does not consider the latency aspect of the state when deciding whether to use it. • C2: Processor power state - This state offers improved power savings over the C1 state. The worst-case hardware latency for this state is provided via the ACPI system firmware and operating software can use this information to determine when the C1 state should be used instead of the C2 state. • C3: Processor power state - This state is not supported. The C3 state offers improved power savings over the C1 and C2 states. The worst-case hardware latency for this state is provided via the ACPI system firmware and the operating software can use this information to determine when the C2 state should be used instead of C3 state. While in the C3 state, the processor's caches maintain state but ignore any snoops.
Cache Line	The unit of memory that is copied to and individually tracked in a cache. Specifically, 64 bytes of data or instructions aligned on a 64-byte physical address boundary.
Cfg	Used as a qualifier for transactions that target PCI configuration address space.
Character	The raw data byte in an encoded system (for example, the 8b value in a 8b/10b encoding scheme). This is the meaningful quantum of information to be transmitted or that is received across an encoded transmission path.
Coherent (C)	Transactions that ensure that the processor's view of memory through the cache is consistent with that obtained through the I/O subsystem. In Chipset 89xx Series PCH, Coherent (C) memory regions are coherent with IA caches when accessed from AIOC agents. Accesses to these memory regions enter the memory system through the IMCH. Memory regions that are coherent with IA caches must be accessible to the IA CPU
Command	The distinct phases, cycles, or packets that make up a transaction. Requests and Completions are referred to generically as Commands.

Table 1-6. Glossary Table (Sheet 2 of 5)

Term	Definition
Completion	A packet, phase, or cycle used to terminate a Transaction on a interface, or within a component. A Completion will always refer to a preceding Request and may or may not include data and/or other information.
Core Power Well	Main system power, turns off in S3 – S5
Cyclic Redundancy Check	A number derived from, and stored or transmitted with, a block of data in order to detect corruption. By recalculating the CRC and comparing it to the value originally transmitted, the receiver can detect some types of transmission errors.
Deasserted	Signal is set to a level that represents logical false.
Deferred Transaction	A processor bus Split Transaction. The requesting agent receives a Deferred Response which allows other transactions to occur on the bus. Later, the response agent completes the original request with a separate Deferred Reply transaction.
Delayed Transaction	A transaction where the target retries an initial request, but unknown to the initiator, forwards or services the request on behalf of the initiator and stores the completion or the result of the request. The original initiator subsequently reissues the request and receives the stored completion.
Direct Memory Access	Method of accessing memory on a system without interrupting the processors on that system.
Downstream	Describes commands or data flowing away from the processor-memory complex and toward I/O. The terms Upstream and Downstream are never used to describe transactions as a whole. (for example, Downstream data may be the result of an Outbound Write, or an Inbound Read. The Completion to an Inbound Read travels Downstream.)
Full Duplex	A connection or channel that allows data or messages to be transmitted in opposite directions simultaneously.
Gb/s	Gigabits per second (10^9 bits per second)
GB/s	Gigabytes per second (10^9 bytes per second)
Global visibility	An operation is said to be globally visible when all side-effects of the operation are visible to every observer in the system. For example, a write to some resource (for example, memory location, control register, etc.) R achieves global visibility when a read of R by all other agents is guaranteed to return the new value.
Gx States	Global system states (Gx states) apply to the entire system and are visible to the user. • G3: Mechanical off - A computer state that is entered and left by a mechanical switch. It is implied by the entry of this off state through a mechanical means that no electrical current is running through the circuitry and that it can be worked on without damaging the hardware or endangering service personnel. • G2/S5: Soft Off - A computer state where the computer consumes a minimal amount of power. • G1: Sleeping - A computer state where the computer consumes a small amount of power, user mode threads are not being executed, and the system “appears” to be off (from an end user’s perspective, the display is off, and so on). • G0: Working - A computer state where the system dispatches user mode (application) threads and they execute. In this state, peripheral devices are having their power state changed dynamically.
Half Duplex	A connection or channel that allows data or messages to be transmitted in either direction, but not simultaneously.
Implicit Writeback	A snoop-initiated data transfer from the bus agent with the modified Cache Line to the memory controller due to an access to that line.

Table 1-6. Glossary Table (Sheet 3 of 5)

Term	Definition
Inbound	A transaction where the request destination is the processor-memory complex and is sourced from I/O. The terms Inbound and Outbound refer to transactions as a whole and never to Requests or Completions in isolation. (for example, an Inbound Read generates Downstream data, whereas an Inbound Write has Upstream data. Even more confusing, the Completion to an Inbound Read travels Downstream.)
Industry Standard Architecture (ISA)	A 16-bit bus architecture associated with the IBM AT motherboard designed to connect motherboard circuitry to expansion card devices that is now considered Legacy.
Initiator	The source of requests. [IBA] An agent sending a request packet on 3GIO is referred to as the Initiator for that Transaction. The Initiator may receive a completion for the Request. [3GIO]
ISA Regime	A special legacy mode to support ISA-based devices which have been integrated into the chipset. It opens a dedicated channel from the peripheral device to the processor bus. While in this mode, the legacy device is granted exclusive accesses to memory and the ability to use Tenured Transactions.
Isochronous	A classification of transactions or a stream of transactions that require service within a fixed time interval.
Lane	A set of differential signal pairs, one pair for transmission and one pair for reception. A by-N Link is composed of N Lanes.
Layer	A level of abstraction commonly used in interface specifications as a tool to group elements related to a basic function of the interface within a layer and to identify key interactions between layers.
Legacy	Functional requirements handed down from previous chipsets, or PC compatibility requirements from the past.
Link	The collection of two Ports and their interconnecting Lanes. A Link is a dual simplex communications path between two components.
LPC Bus	Low Pin Count connection used to connect to the super I/O device.
Master	A device or logical entity that is capable of initiating transactions. A Master is any potential Initiator.
Mbyte/s	Megabytes per second (10^6 bytes per second)
Mem	Used as a qualifier for transactions that target memory space. (for example, a Mem read to I/O.)
Metastability	A characteristic of flip flops that describes the state where the output becomes non-deterministic. Most commonly caused by a setup or hold time violation.
Multi Media Timer (MMT)	See High Precision Event Timer (HPET) in Table 1-5.
Non-Coherent	Transactions that may cause the processor's view of memory through the cache to be different than that obtained through the I/O subsystem.
North	Usually refers to bridges. The bridge or device that is closer to the processor-memory complex.
Ordering	Refers to the order in which signals and/or memory accesses to different locations must reach global visibility to ensure some behavior. This excludes the "ordering" necessary to prevent data hazards which are accesses to the same location.
Outbound	A transaction where the request destination is I/O and is sourced from the processor-memory complex. The terms Inbound and Outbound refer to transactions as a whole and never to Requests or Completions in isolation. (for example, an Outbound Read generates Upstream data, whereas an Outbound Write has Downstream data. Even more confusing, the Completion to an Outbound Read travels Upstream.)
OWord	128 bits of data on a naturally aligned sixteen-byte boundary (for example, the least significant four bits of the byte address are b"0000"). This is the native size of the IMCH datapath.

Table 1-6. Glossary Table (Sheet 4 of 5)

Term	Definition
Packet	The indivisible unit of data transfer and routing, consisting of a header, data, and CRC.
PCI Reset	PCIRST#. This is the secondary PCI Bus reset signal. It is a logical OR of the primary interface PLTRST# signal and the state of the Secondary Bus Reset bit of the Bridge Control register (D30:F0:Reg3Eh[6]).
Peer-to-Peer	Transactions that occur between two devices independent of memory or the processor.
Platform Reset	ILB asserts PLTRST# to reset devices that reside on the primary PCI bus. The ILB asserts PLTRST# during power-up and when a hard reset sequence is initiated through the CF9h register. PLTRST# is driven inactive a minimum of 1 ms after both PWROK and VGATE are driven high. PLTRST# is driven for a minimum of 1 ms when initiated through the CF9h register.
Plesiochronous	From Greek, meaning almost synchronous. Describes signals that have the same nominal digital rate, but are synchronized on different clocks. Any variation in rate is constrained within specified limits, which allows a device to process the data signal without buffer underflow or overflow by making periodic compensating adjustments that repeat or delete dummy data bits. However, there is no limit to the phase difference that can accumulate between the signals over time.
Port	• Logically, an interface between a component and a PCI Express Link. • Physically, a group of Transmitters and Receivers located on the same chip that define a Link.
Posted	A Transaction that is considered complete by the initiating agent or source before it actually completes at the Target of the Request or destination. All agents or devices handling the Request on behalf of the original Initiator must then treat the Transaction as being system visible from the initiating interface all the way to the final destination. Commonly refers to memory writes.
Push Model	Method of messaging or data transfer that predominately uses writes instead of reads.
Queue	A first-in first-out (FIFO) structure.
Receiver	• The Agent that receives a Packet across an interface regardless of whether it is the ultimate destination of the packet. • More narrowly, the circuitry required to convert incoming signals from the physical medium to more perceptible forms.
Request	A packet, phase, or cycle used to initiate a Transaction on a interface, or within a component.
Reserved	The contents or undefined states or information that are not defined at this time. Using any reserved area is not permitted. Reserved register bits must be set to 0. However, when stated, there may be specific instances where a reserved register is either non-zero, or there may be a requirement to make it non-zero.
Resume Power Well	Trickle from power supply, only turns off when power is disconnected from wall.
Resume Reset	Signal that resets the parts of the PCH in the resume power well, generated when the trickle supply turns on.
RTC Power Well	Powered by a coin cell battery and only turns off when the battery is drained. Powers the RTC and some resume events.

Table 1-6. Glossary Table (Sheet 5 of 5)

Term	Definition
Sx States	Sleeping states (Sx states) are types of sleeping states within the global sleeping state, G1. • S5: Soft Off state. The main memory power plane is shut down in addition to the clock synthesizer and core well power planes for the processor and CMI. The CMI resume well is still powered. • S4: Sleeping state - This state is only used to transition to or from the S5 state. **The S4 state is not a supported power management state in** CMI. • S3: Suspend to RAM (STR) state - The clock synthesizer and core well power planes for the processor and CMI are shut down, but the main memory power plane and the CMI resume well remain active. All clocks from synthesizers are shut down during the S3 state. • S0: Awake state - Power Management state when all power planes are active.
Simplex	A connection or channel that allows data or messages to be transmitted in one direction only.
SMBus	System Management Bus. A two-wire interface through which various system components may communicate.
Snooping	A means of ensuring cache coherency by monitoring all memory accesses on a common multi-drop bus to determine if an access is to information resident within a cache.
South	Usually refers to bridges. The bridge or device that is further from the processor-memory complex.
South Port	The PCI Express downstream root port(s) on the ILB.
Split Lock Sequence	A sequence of transactions that occurs when the target of a lock operation is split across a processor bus data alignment or Cache Line boundary, resulting in two read transactions and two write transactions to accomplish a read-modify-write operation.
Split Transaction	A transaction that consists of distinct Request and Completion phases or packets that allow use of bus, or interconnect, by other transactions while the Target is servicing the Request.
Symbol	An expanded and encoded representation of a data Byte in an encoded system (for example, the 10b value in a 8b/10b encoding scheme). This is the value that is transmitted over the physical medium.
Symbol Time	The amount of time required to transmit a symbol.
Target	A device that responds to bus Transactions. The agent receiving a request packet is referred to as the Target for that Transaction.
Tenured Transaction	A transaction that holds the bus or interconnect until complete, effectively blocking all other transactions while the Target is servicing the Request.
Transaction	An overloaded term that represents an operation between two or more agents that can be comprised of multiple phases, cycles, or packets.
Transaction Identifier	A multi-bit field used to uniquely identify a transaction. Commonly used to relate a Completion with its originating Request in a Split Transaction system.
Transmitter	• The Agent that sends a Packet across an interface regardless of whether it was the original generator of the packet. • More narrowly, the circuitry required to drive signals onto the physical medium.
Upstream	Describes commands or data flowing toward the processor-memory complex and away from I/O. The terms Upstream and Downstream are never used to describe transactions as a whole. (for example, Upstream data may be the result of an Inbound Write, or an Outbound Read. The Completion to an Outbound Read travels Upstream.)

§ §

2.0 Architecture Overview

2.1 Introduction

The PCH targets communications and embedded platforms. These product's stringent performance, power, and cost targets can be met by integrating common platform components, adding on-chip hardware accelerators.

Systems based on the PCH include four major blocks:

- One or more IA CPU cores
- IA-compatible memory and I/O controller hubs
- On-chip memory and controllers for external memory
- A model-specific communications complex

Examples of such communications-oriented functions are:

- PCI Express target
- Ethernet
- Protocol acceleration hardware:
  - — Intel® QuickAssist Integrated Accelerators
  - — Security accelerationor bulk encryption, hashing, public/private key generation, and compression.

From a system standpoint, Chipset 89xx Series can be most easily described as a PCH that includes both standard PC interfaces (for example, PCI Express Root Complex, SATA*, USB*, etc.) along with Intel® QuickAssist Technology and GbE interfaces.

The PCH can be accessed by the IA processor via two interfaces: a DMI interface that provides connectivity to the standard PC interfaces and a PCI Express interface that provides connectivity to the PCH's PCI Express End Point (EP) for Intel® QuickAssist Technology and access to the GbE MACs. These interfaces are not dependent on one another and can be used independently in a system. This offers three modes in which the PCH can be used:

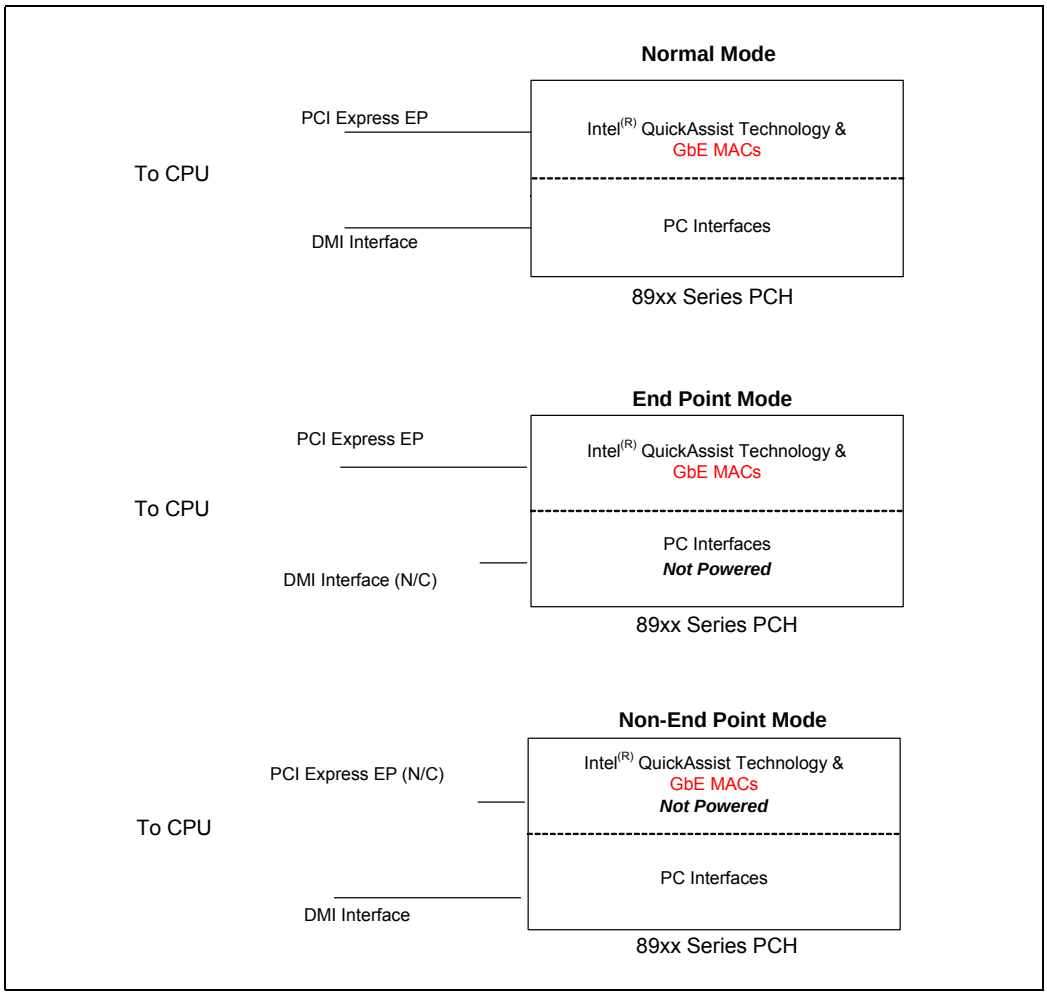
- Normal Mode (via DMI and PCI Express EP Interfaces)
  For PC Interfaces, GbE & Intel® QuickAssist Technology
- End Point Mode (via PCI Express EP Interface Only)
  For GbE & Intel® QuickAssist Technology
- Non-End Point Mode (via DMI Interfaces Only)
  For PC Interfaces

Note: GbE MAC functionality is SKU dependant.

In this document:

- *Chipset*, *Platform Controller Hub*, and *PCH* are used as generic references to the Intel® Communications Chipset 89xx Series.
- Intel® Communications Chipset 89xx Series.
  - DH8900CC includes Chipset 89xx Series SKUs where 8900 ≤ *SKU* ≤ 8920
  - DH8900CL includes Chipset 89xx Series SKUs where 8925 ≤ *SKU* ≤ 8955

Figure 2-1. PCH Modes



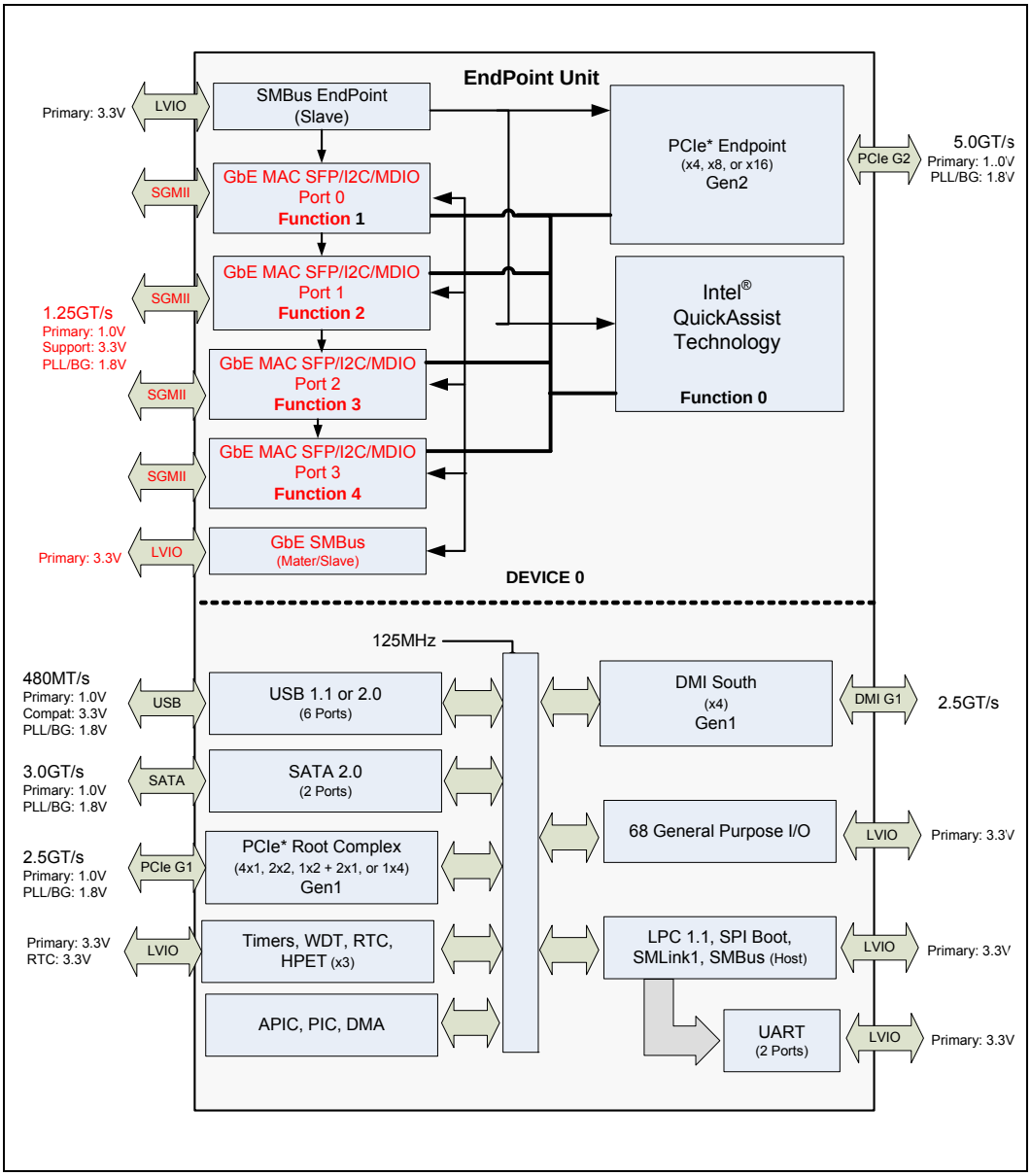
Note: Functions in **RED** are not available on the DH8900CL devices.

2.2 PCH Architecture Overview

This section provides an overview of the PCH architecture. Section 2.2.1, "PCH Block Summary" gives a high-level summary for each of the major blocks and their internal interfaces. Section 2.2.2, "PCH External Interfaces" reviews PCH's external chip interfaces and internal block gear ratios.

The following figure shows the major PCH blocks.

Figure 2-2. PCH Block Diagram



Note: Functions in **RED** are **NOT** available in the DH89xxCL devices.

2.2.1 PCH Block Summary

The PCH has two major interfaces to the IA processor: the PCIe* interface for Intel® QuickAssist Technology and GbE MACs and the DMI interface or standard PCIO interfaces.

2.2.2 PCH External Interfaces

Table 2-1 summarizes the key features of PCH's external interfaces. Specific products may elect to make a subset of these interface visible based on their needs and requirements.

Table 2-1. PCH External Interface Summary

Name	Qty.	Description
PCI Express* (Gen2, Endpoint)	1	Supports x1, x4, x8, or x16 widths[1] and provides the primary interface into the PCH's PCIe* Endpoint for GbE and services. This interface is Gen2 capable. The PCH has no parallel PCI interface.
DMI 2.0 Gen1	1	Supports x4 width and provides the primary interface into the PCH's Legacy I/O.
Gigabit Ethernet	4[a]	10/100/1000 Gigabit Ethernet[2] MACs with SGMII interface. Each of the ports support IEEE 1588 time synchronization.
MDIO	4[a]	MDIO/I2C support the configuration of PHY or SFP.
PCI Express Gen1 (root)	1-4	Supports 4x1, 2x2, 1x2 + 2x1, or 1x4 configurations. This interface is PCI Express 2.0 Compliant, operating at Gen1 speed (2.5 GT/s). The PCH has no parallel PCI interface.
USB* 2.0	1	Universal Serial Bus 2.0 host controller interface, supports six USB* ports (shared with USB* 1.1 ports)
USB* 1.1	1	Universal Serial Bus 1.1 host controller interface, supports six USB* ports (shared with USB* 2.0 ports)
SATA*	2	SATA* 1.0 or 2.0 used to attached external hard drives.
LPC	1	Low Pin Count Bus (LPC) interface.
SPI	1	Serial Peripheral Interface (SPI). Used for boot device.
GPIO	20[3]	Programmable General Purpose I/O (GPIO) pins. Of the pins, many have alternate functions defined.
SMBus/I2C	2	I2C compatible SMBus2.0 connections.
UART	2	16550 compatible asynchronous serial ports that support data rate of at least 115Kbits/sec.

1. SKU dependent
2. See the Supported Ethernet PHY Device for the PCH.
3. Dedicated.

2.2.3 IA Compatibility

100% IA platform compatibility allows applications and operating systems to scale-down from mainstream IA products to Chipset 89xx Series-based products. The ability to deploy an unmodified binary IA operating system with existing IA device drivers for third party PCI Express devices on a Chipset 89xx Series-based product is critical.

From an operating system developer's perspective, a Chipset 89xx Series-based product manifests itself like a new IA chipset and not as an entirely different platform. That is, devices can be enumerated and configured via existing PCI configuration mechanisms, signaling and ordering follow established PCI rules, expected legacy IA chipset features such as interrupt and DMA controllers, timers, real-time clock and ACPI power management interfaces are supported. While most communications and embedded applications live on modern 32-bit operating systems (Linux, BSD, VxWorks, QNX, etc.), support for 8/16 bit environments seems unnecessary. However, many users of the IA legacy functionality exist.

§ §

3.0 PCH Platform Memory and Device Configuration

3.1 Overview

This chapter presents the views of the major address spaces and device configuration structures as seen by various internal and external agents in a PCH system. Three related aspects are covered:

- The memory maps seen by various internal and external agents in a PCH system.
- The endianess seen by various agents in a PCH system and mechanisms PCH uses to allow communication between agents with different endianess expectations.
- The PCI configuration infrastructure for the PCH, which the PCH exposes through its memory maps.

3.1.1 Configuration Objectives

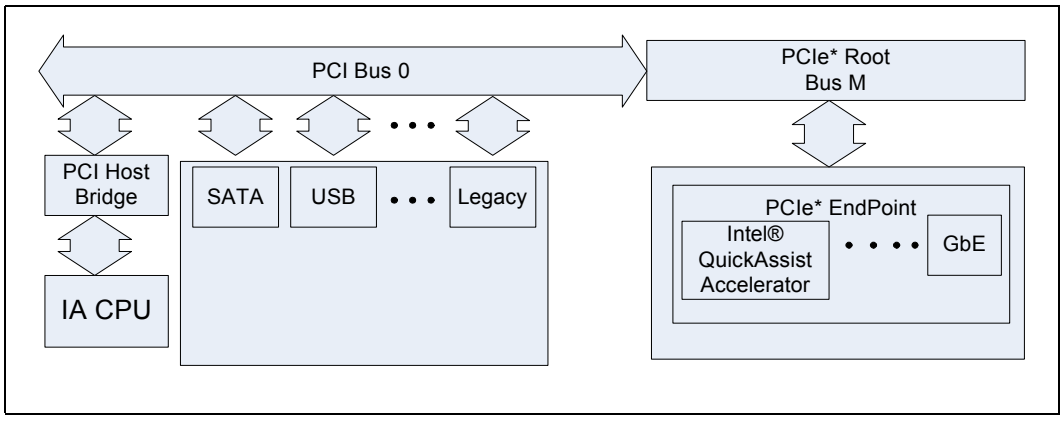
The PCH device and configuration model blends the architectures of many disparate components into a unified whole. The major goals for the device configuration and access architecture include:

- Provide a configuration and access model that is aligned with existing IA platform algorithms.
- Support a unified address space model.

The IA is the primary agent responsible for device configuration. This is true across all supported SKU configurations.

To provide device configuration and operation capabilities that are aligned with the IA platform, the PCH exposes the devices through PCI devices or functions.

Figure 3-1. Device-Centric Logical View of PCH Devices



In this approach, the PCH extends the hardware blocks that make up the PCIe* Endpoint to allow them to materialize as a collection of PCI functions. This allows the IA to use existing system software to discover, enumerate, and operate all devices in the chip. The PCIe* Endpoint is not built around a PCI fabric, but rather, layers PCI abstractions on top of existing infrastructure.

3.1.2 Terminology and Conventions

Throughout this chapter, the generic term "device" refers to either a PCI device or a function of a PCI device. The text will be explicit when the distinction between device and function is important.

Addresses are always in hexadecimal and broken into 16-bit segments, such as `0_FEED_BEEF`. When the distinction is important and not obvious, addresses are subscripted with "V", "P", or "S" for virtual, physical, or system address spaces.

3.2 IA Platform Infrastructure

The PCH provides an IA platform infrastructure with respect to endianess, address spaces and memory maps, configuration, etc.

This section focuses on the IA views and expectations around the endianess, address spaces and memory maps, and configuration for a basic IA platform. The PCH operates within this framework.

Later sections in this chapter and volumes discuss the specifics of the PCH implementation. These discussions highlight how and where the PCH differs from the framework.

3.2.1 General IA Platform View of the Physical Address Space

The PCH operates within a standard IA physical address space. To support specific processors, the PCH supports at least 46-bit physical address spaces. For more information on the specific layout of the address space, consult the appropriate IA documentation for the core that is used with the PCH.

3.2.2 IA Platform View of Configuration

The PCH is comprised of IA functional blocks that are exposed through a PCI infrastructure. The PCH extends this PCI infrastructure to expose the functionality in the PCIe* Endpoint, as described in Section 3.6, "PCI Configuration" on page 106.

Before describing how the PCIe* Endpoint integrates with the IA-based PCH blocks, it is helpful to consider how PCI exposes PCH functionality. Logically, the software-visible sub-blocks of the PCH materialize as PCI devices and functions on PCI bus 0 of the system through three independent address spaces:

- Configuration Space: Each function of each device has at least 256B of configuration space that is mapped to a fixed location based on bus, device and function number (PCI Express devices can provide for larger configuration spaces). This space provides system software with basic information on the device and allows for device-independent configuration.

- Memory-Mapped I/O (MMIO) and I/O Spaces: Each function of each device can request up to six MMIO or I/O regions of device-specified sizes to be mapped into physical address space through base address registers in the configuration header. System software selects the base address of each region. These spaces support device-specific operation such as access to device-specific control registers.

In general, the IA uncore claims configuration accesses (for example, those accesses that target configuration space) to some devices on bus 0 and routes configuration accesses to the remaining devices to the PCH over the DMI interface using Type 0 PCI configuration transactions.

3.3 High-Level Views

This section presents an overview of some of the general characteristics of the agents that the various PCH memory maps expose.

3.3.1 Characteristics of External System Memory (DRAM)

The PCH operates in the standard IA physical address space. See the relevant IA CPU documentation for more details on address space size and layout.

3.3.2 Characteristics of Internal and External Memories

Table 3-1 defines the supported operations by memory type. The table uses the following notation to indicate the behavior of the PCH:

- "–" means the operation is not supported by the PCH.
- "S" implies that the operation happens as a single atomic[1] update to memory. In other words, either the update is observable in its entirety or not at all.
- "M" implies that the operation may happen as multiple updates to memory. In other words, other agents can observe different parts of the affected memory location change values in any order but the end state of the memory location will be the desired value. This "flickering lights" effect makes such memory accesses useless for multi-agent synchronization unless a semaphore or flag variable is used to guard access to the shared location[2].

This table only applies to aligned-to-size operations; that is, a 4-byte operation is aligned to a 4-byte boundary, an 8-byte operation is aligned to an 8-byte boundary, etc.)

1. In the sense that it cannot be divided into multiple smaller writes.
2. In guarding the location, visibility of the new flag must imply that the "flickering" has stopped.

Table 3-1. Supported Operations by Memory Type

Operation		IA CPU		
Type	Size [Bytes]	IA WB Cacheable	IA UC	MMIO
Read, Write	1	S	S	S
	2	S	S	S
	4	S	S	S
	8	S	S	S
	16	S	M	M
	32	–	–	–
	64	S	–	–
	128	–	–	–
Atom Read-Modify-Write (Semaphore)	1	S	M	–
	2	S	M	–
	4	S	M	–
	8	S	M	–

3.3.3 Characteristics of Device Configuration

To be able to leverage existing IA BIOS, Operating Systems, and power management software, PCH's configuration mechanisms follow existing IA platform approaches. Of the two major complexes of the PCH:

- The PCH can use normal IA platform configuration algorithms.
- To interoperate with normal IA platform configuration algorithms, the PCIe* Endpoint must be configured by the IA processor.

To the extent possible, the configuration algorithms should be independent of the specific topology to allow for maximum re-use of hardware and software designs. Further, they should be forward-looking to support future platforms with address spaces larger than 32 bits. This implies that control registers should be sized to allow the device to perform an access anywhere in a 64-bit address space. The goal does not require the device to be able to be a 64-bit target (for example, it need not support 64-bit BARs to allow its MMIO regions to materialize in PCI H).

With the PCH, the boot and configuration process is:

1. The IA boots from a FLASH device on the PCH SPI interface.
2. System software discovers and configures the devices and function on PCI bus 0 in the PCH and PCIe* Endpoint.
3. System software configures other buses on the system.

Once this process completes, the PCH and IA are ready for operation.

3.4 Memory Map for PCIe* Endpoint-Attached Devices

All PCIe* Endpoint-attached agents support independent target IDs that provide independent address spaces. Agents that do not natively support 64-bit addressing are mapped into the full 64-bit address space as defined in the following sections.

Table 3-2. Address Space Sizes of PCIe* Endpoint-Attached Devices

Address Space Size [b]	Devices
50	Gigabit Ethernet MACs, PCIe* Endpoint

All addresses are zero-extended to 64-bits by definition.

3.5 PCH Endianness

The PCH operates in an IA platform environment that is little-endian.

3.6 PCI Configuration

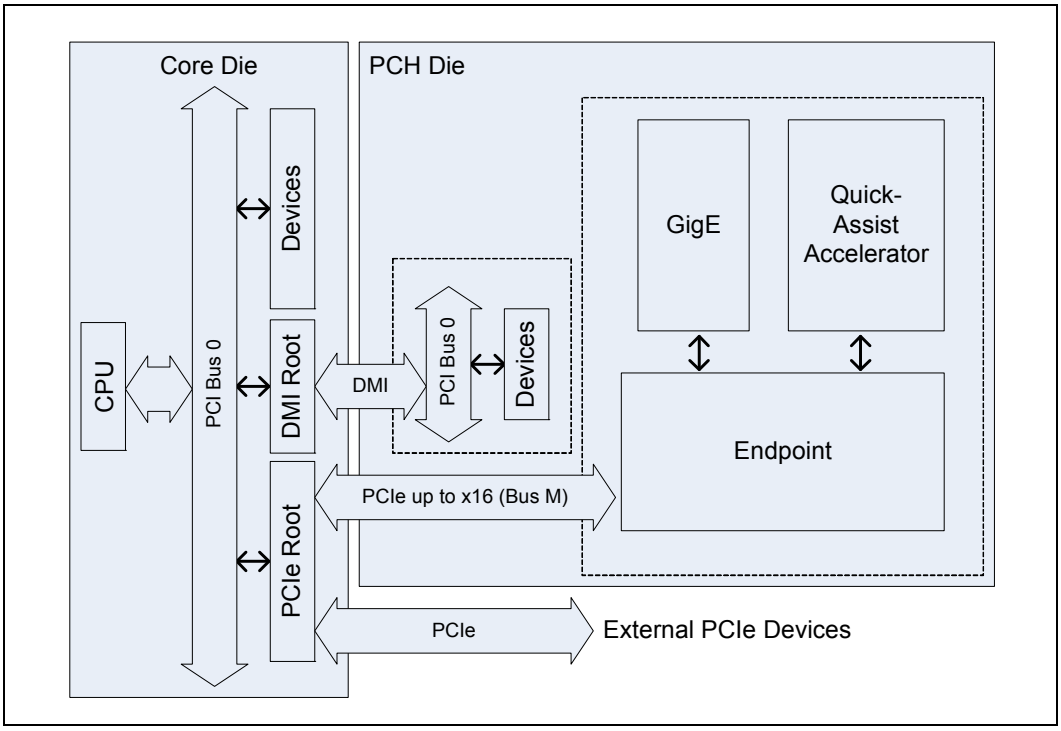
This section presents an overview of the implementation that integrates the PCIe* Endpoint with the IA PCI infrastructure for configuration.

- PCI mechanisms (configuration space, memory-mapped I/O spaces, and I/O spaces) expose state for configuration.
- The IA-32 core performs all system configuration and initialization.
- The IA-32 core configures the PCIe* Endpoint using standard IA platform algorithms (for example, PCI-based discovery, enumeration, and configuration) with modifications for the specific mix of functionality that the PCH instantiations provide.
- The GbE MACs and other sub-blocks appear as PCI functions of a PCI Endpoint and allow the use of the standard PCI discovery, enumeration, and configuration algorithms (this implies, for example, corresponding PCI configuration headers, etc.).
- A PCH-specific user driver handles interaction with external non-PCI agents attached to PCH through its PCIe* Endpoint I/O interfaces. That is, the PCH user driver, not BIOS, will "discover" and operate any such devices. Since these devices do not implement PCI semantics, it is expected that they will not allocate MMIO regions beyond those already allocated for the PCI view of the appropriate PCIe* Endpoint function.
- The O/S always allocates an aperture in the memory map for any PCI device that defines one or more BARs, even if the device is unknown to the O/S at discovery.

3.6.1 Overview

The PCH integrates the legacy PC Interfaces (DMI section) and PCIe* Endpoint into the PCI fabric as Figure 3-2 shows. This figure presents a ***logical*** view of the system and does not show exact physical connectivity nor the exact fabrics.

Figure 3-2. Attaching PCH to the PCI Fabric (Logical Perspective)



In the figure above, PCI bus 0 originates in the CPU die and reaches internal PCI devices through internal paths. Bus 0 is bridged into the PCH via an DMI interconnect and into the PCIe* Endpoint via a subset of the PCI Express lanes on the CPU die. The PCI Express Endpoint materializes the individual functions of the endpoint device on bus M as the figure illustrates.

3.6.2 Device Tree

The PCIe* Endpoint implements the function semantics, effectively mediating accesses between PCI bus M and its functions.

Devices and functions can request space in the system memory and I/O address maps through BARs in the configuration header. In general, the PCH materializes most device control and status registers in memory-mapped regions allocated by a BAR. The only exception lies in the standard PCI configuration, status, and capability registers that PCI requires which materialize only in PCI configuration space.

The remainder of this section summarizes the device tree that the PCH implements. This summary includes a mapping between PCI devices and the PCH blocks along with the value of the device ID, class code, and a summary of the resources (for example, registers, memory, etc.) that each device requests.

The following table summarizes the PCI devices that the PCH materializes.

Table 3-3. PCH PCI Device Summary

Device Name	PCI		
	B/D/F[1]	DH89xxCC Device ID	DH89xxCL Device ID
PCIe* Function Subtractive decode	B0:D28:Fn	0x244E	0x244E
PCIe* Function Subtractive decode	B0:D28:Fn	0x2448	0x2448
LPC Interface	B0:D31:F0	0x2310	0x2390
SATA* Controller #1: AHCI, supporting 2 ports	B0:D31:F2	0x2323	0x23A3
SATA* Controller #2: IDE, supporting 2 ports	B0:D31:F5	0x2326	0x23A6
SMBus Controller	B0:D31:F3	0x2330	0x23B0
Thermal Subsystem	B0:D31:F6	0x2332	0x23B2
USB* 2.0 Controller	B0:D29:F0	0x2334 0x2335	0x23B4 0x23B5
PCI-Express Root Port #1	B0:D28:F0	0X2342 0x2343	0x23C2 0x23C3
PCI-Express Root Port #2	B0:D28:F1	0x2344 0x2345	0x23C4 0x23C5
PCI-Express Root Port #3	B0:D28:F2	0x2346 0x2347	0x23C6 0x23C7
PCI-Express Root Port #4	B0:D28:F3	0x2348 0x2349	0x23C8 0x23C9
WDT Timer for per core reset	B0:D31:F7	0x2360	0x23E0
Intel® Management Engine Interface #1 (MEI #1)	B0:D22:F0	0x2364	0x23E4
Intel® Management Engine Interface #2 (MEI #2)	B0:D22:F1	0x2365	0x23E5

1. PCI bus number, device number, and function number.

Table 3-4. EP PCI Device Summary

Device Name	PCI		
	B/D/F[1]	DH89xxCC Device ID	DH89xxCL Device ID
PCIe* Endpoint & IQAT	BM:DD:F0	0x0434	0x0435
GbE, Port 0	BM:DD:F1	0x0436 0x0438 0x043A 0x043C 0x0440[2]	N/A
GbE, Port 1	BM:DD:F2		
GbE, Port 2	BM:DD:F3		
GbE, Port 3	BM:DD:F4		

1. PCI bus number, device number, and function number.
2. All GbE functions use one of the DIDs at a time. A DID is assigned to the GbEs as shown in Table 3-5.

Table 3-5. DH89xxCC GbE PCI Device ID Summary

DH89xxCC GbE DID	Usage
0x0436	Default DID – this is not handled in the OS network driver.
0x0438	DH8900CC Series Gigabit Network Connection.
0x043A	DH8900CC Series Gigabit Fiber Network Connection.
0x043C	DH8900CC Series Gigabit Backplane Network Connection.
0x0440	DH8900CC Series Gigabit SFP Network Connection.

Table 3-3 and Table 3-5 imply a logical view of the configuration registers. They are not meant to imply physical location.

3.6.3 Materializing Device Structures

The PCH exposes PCIe* Endpoint resources through standard PCI abstractions: devices/functions, configuration spaces and memory-mapped I/O spaces.

Access to MMIO and I/O spaces are provided through memory and I/O read/write instructions, respectively. The addressing of these spaces for a given device depends on the specific mapping that the PCI configuration header establishes through BARs.

PCI defines two mechanisms for accessing the 256B of each device/function configuration registers located in PCI configuration space.

- PCI Mechanism: The header is accessed using 1-, 2-, or 4-byte `IN` and `OUT` instructions that access the PCI configuration address and data I/O ports at addresses `0CF8h - 0CFBh` and `0CFCh - 0CFFh`, respectively, in the IA I/O space. This mechanism allows access only to the 256B PCI-compatible configuration space.
- PCI Express Enhanced Mechanism: The header is accessed using 1-, 2-, or 4-byte memory accesses to the 256MB region starting at HECBASE. This mechanism allows access to an expanded 4KB configuration space that PCI Express defines (the first 256B are, by definition, the PCI-compatible configuration space).

The PCH supports both mechanisms. These mechanisms differ in the address space they use to access the header. The PCI mechanism travels through IA I/O space while the PCI Express Enhanced mechanism travels through IA memory space. The address format that the mechanisms use is identical to the standard IA platform format that encodes the PCI bus, device, and function numbers along with a register offset or number.

For either access method, the hardware in the PCIe* Endpoint that implements the configuration headers must be able to process accesses of the appropriate sizes.

3.6.4 PCI Configuration Headers

The PCI specification requires each PCI function to provide a 256B configuration space. The first 64B of this space contains a standard PCI configuration header and the remaining 192B contains any device-specific registers, capabilities records, etc. needed by the function. There are two flavors of configuration headers:

- All non-bridge devices provide a PCI type 0 configuration headers. This form of header is used to represent devices on the PCI fabric.
- All bridge devices provide a PCI type 1 configuration header. This form of header is used to represent bridge devices in the PCI fabric.

Since the PCIe* Endpoint devices that the PCH exposes through PCI are not, strictly speaking, PCI devices or functions. The following tables describe the support in greater detail.

Table 3-6 summarizes the fields in a PCI type 0 header (for example, header for non-bridge devices) and identifies which fields the PCH implements for Endpoint functions. The PCH hardware implements the appropriate PCI semantics for all supported registers and fields in this table.

Table 3-6. PCI Configuration Header Support for Type 0 Headers in PCIe* Endpoint Devices (Sheet 1 of 2)

Offset	Register and Field		Bit(s)	Supt.[1]	Acc.[2]	Notes
00h - 01h	Vendor ID		15:0	Y	RO	Required by PCI.
02h - 03h	Device ID		15:0	Y	RO	Required by PCI.
04h - 05h	Command Register	Interrupt Disable	10	Y	RW	Supported in devices that can use INTx[3].
		Fast Back-to-Back Enable	9	N	RO	Not supported.
		SERR# Enable	8	Y	RW	Supported.
		Parity Error Response	6	Y	RW	Supported.
		VGA Palette Snoop	5	N	RO	Not supported.
		Mem. Write & Inval. Enable	4	N	RO	Not supported.
		Special Cycles	3	N	RO	Not supported.
		Bus Master Enable	2	Y	RW	QAT, GbE, devices.
		Memory Space Enable	1	Y	RW	All devices: QAT, GbE.
		I/O Space Enable	0	Y	RW	For GbE; GbE materializes in I/O space.
				N	RO	All devices: QAT except GbE.
06h - 07h	Status Register	Detected Parity Error	15	Y	RW1C	Supported.
		Signalled System Error	14	Y	RW1C	Supported.
		Received Master-Abort	13	Y	RW1C	Supported.
		Received Target-Abort	12	Y	RW1C	Supported.
		Signalled Target-Abort	11	Y	RW1C	Supported.
		DEVSEL Timing	10:9	N	RO	Not supported.
		Master Data Parity Error	8	Y	RW1C	Supported.
		Fast Back-to-Back Capable	7	N	RO	Not supported.
		66MHz Capable	5	N	RO	Not supported.
		Capabilities List	4	Y	RO	Setup based on capabilities exposure by device.
		Interrupt Status	3	Y	RO	Supported.
08h	Revision ID		7:0	Y	RO	Required by PCI.
09h - 0Bh	Class Code		23:0	Y	RO	Required by PCI.
0Ch	Cache Line Size		7:0	N	RO	Not supported.

Table 3-6. PCI Configuration Header Support for Type 0 Headers in PCIe* Endpoint Devices (Sheet 2 of 2)

Offset	Register and Field	Bit(s)	Supt.[1]	Acc.[2]	Notes
0Dh	Latency Timer	7:0	N	RO	Not supported.
0Eh	Header Type	7:0	Y	RO	Required by PCI.
0Fh	BIST	7:0	N	RO	Not supported.
10h - 27h	Base Address (x6)	6 x 31:0	Y	RW	Devices that materialize in I/O or memory spaces will populate these slots as necessary based on address space needs.
28h - 2Bh	CIS Pointer	31:0	N	RO	Not supported.
2Ch - 2Dh	Subsystem VID	15:0	Y	RO	Required by PCI.
2Eh - 2Fh	Subsystem ID	15:0	Y	RO	Required by PCI.
34h	Capability Pointer	7:0	Y	RO	Setup based on capabilities exposure by device.
3Ch	Interrupt Line	7:0	Y	RW	Supported in devices that can use INTx
3Dh	Interrupt Pin	7:0	Y	RO	Supported in devices that can use INTx
3Eh	Min_Gnt	7:0	N	RO	Not supported.
3Fh	Max_Lat	7:0	N	RO	Not supported.

1. Supported fields provide appropriate PCI semantics. Unsupported fields always return zero on reads unless otherwise noted.
2. RW1, RO and RW access types indicate that the register or field supports read-only access and read/write access

The specific portion of the 256B PCI configuration space that is active in a device depends on the needs of the specific device. In general, a device requires far less than 256B of storage to implement a typical configuration space. Regions of the 256B configuration space that are not required are reserved and need only support default behavior compliant with the PCI specification:

- All PCI devices must treat Configuration Space write operations to reserved registers as no-ops; that is, the access must be completed normally on the bus and the data discarded. Read accesses to reserved or unimplemented registers must be completed normally and a data value of 0 returned.

§ §

4.0 Functional Description

This chapter describes system functions and interfaces.

4.1 PCI Express* Root Ports

There are four PCIe* Root Ports (1 through 4). The root ports operate at Gen1 speed (2.5GT/s) but are compliant to PCIe* Gen2 specification Messaging Protocol. The ports all reside in bus 0:device 28, and take function 0 – 3. Port 1 is function 0, port 2 is function 1, port 3 is function 2, and port 4 is function 3.

The PCI Express* Root Ports can be independently configured to support four x1s, two x2s, one x2 + two x1 or one x4 port widths. The port configuration is set by soft straps in the Flash Descriptor.

4.1.1 Interrupt Generation

The root port generates interrupts on behalf of hot-plug and power management events, when enabled. These interrupts can be pin based or can be MSIs, when enabled.

An interrupt can be generated via a legacy mechanism that is based on the setting of the chipset configuration registers. Specifically, the chipset configuration registers used are the D28IP (Base address + 310Ch) and D28IR (Base address + 3146h) registers.

The following table summarizes interrupt behavior for MSI and wire-modes. Bits refer to the hot-plug and PME interrupt bits.

Table 4-1. MSI vs. PCI IRQ Actions

Interrupt Register	Wire-Mode Action	MSI Action
All bits 0	Wire inactive	No action
One or more bits set to 1	Wire active	Send message
One or more bits set to 1, new bit gets set to 1	Wire active	Send message
One or more bits set to 1, software clears some (but not all) bits	Wire active	Send message
One or more bits set to 1, software clears all bits	Wire inactive	No action
Software clears one or more bits, and one or more bits are set on the same clock	Wire active	Send message

4.1.2 Power Management

4.1.2.1 S3/S4/S5 Support

Software initiates the transition to S3/S4/S5 by performing an IO write to the Power Management Control register. After the IO write completion has been returned to the processor, each root port sends a PME_Turn_Off TLP (Transaction Layer Packet) message on its downstream link. The device attached to the link eventually responds with a PME_TO_Ack TLP message followed by sending a PM_Enter_L23 DLLP (Data Link Layer Packet) request to enter the L2/L3 Ready state. When all of the root ports links are in the L2/L3 Ready state, the power management control logic proceeds with the entry into S3/S4/S5.

Prior to entering S3, software is required to put each device into $D3_{HOT}$. When a device is put into $D3_{HOT}$ it initiates entry into a L1 link state by sending a PM_Enter_L1 DLLP. Thus under normal operating conditions when the root ports sends the PME_Turn_Off message the link is in state L1. However, when the root port is instructed to send the PME_Turn_Off message, it sends it whether or not the link was in L1. Attached EndPoints can make no assumptions about the state of the link prior to receiving a PME_Turn_Off message.

4.1.2.2 Resuming from Suspended State

The root port contains enough circuitry in the suspend well to detect a wake event through the WAKE# signal and to wake the system. When WAKE# is detected asserted, an internal signal is sent to the power management controller to cause the system to wake up. This internal message is not logged in any register, nor is an interrupt/GPE generated due to it.

4.1.2.3 Device Initiated PM_PME Message

When the system has returned to a working state from a previous low power state, a device requesting service sends a PM_PME message continuously, until acknowledge by the root port. The root port takes different actions depending upon whether this is the first PM_PME has been received, or whether a previous message has been received but not yet serviced by the operating system.

If this is the first message received (RSTS.PS - B0:D28:F0/F1/F2/F3:Offset 60h:bit 16 is cleared), the root port sets RSTS.PS, and log the PME Requester ID into RSTS.RID (B0:D28:F0/F1/F2/F3:Offset 60h:bits 15:0). If an interrupt is enabled via RCTL.PIE (B0:D28:F0/F1/F2/F3:Offset 5Ch:bit 3), an interrupt is generated. This interrupt can be either a pin or an MSI if MSI is enabled via MC.MSIE (B0:D28:F0/F1/F2/F3:Offset 82h:bit 0). See Section 4.1.2.4 for SMI/SCI generation.

If this is a subsequent message received (RSTS.PS is already set), the root port sets RSTS.PP (B0:D28:F0/F1/F2/F3:Offset 60h:bit 17) and log the PME Requester ID from the message in a hidden register. No other action is taken.

When the first PME event is cleared by software clearing RSTS.PS, the root port sets RSTS.PS, clear RSTS.PP, and move the requester ID from the hidden register into RSTS.RID.

If RCTL.PIE is set, an interrupt is generated. If RCTL.PIE is not set, a message is sent to the power management controller so that a GPE can be set. If messages have been logged (RSTS.PS is set), and RCTL.PIE is later written from a 0 to a 1, and interrupt is generated. This last condition handles the case where the message was received prior to the operating system re-enabling interrupts after resuming from a low power state.

4.1.2.4 SMI/SCI Generation

Interrupts for power management events are not supported on legacy operating systems. To support power management on non-PCI Express* aware operating systems, PM events can be routed to generate SCI. To generate SCI, MPC.PMCE must be set. When set, a power management event causes SMSCS.PMCS (B0:D28:F0/F1/F2/F3:Offset DCh:bit 31) to be set.

BIOS workarounds for power management can be supported by setting MPC.PMME (B0:D28:F0/F1/F2/F3:Offset D8h:bit 0). When this bit is set, power management events sets SMSCS.PMMS (B0:D28:F0/F1/F2/F3:Offset DCh:bit 0), and SMI # is generated. This bit is set regardless of whether interrupts or SCI is enabled. The SMI# may occur concurrently with an interrupt or SCI.

4.1.3 SERR# Generation

SERR# may be generated through PCI Express mechanisms involving bits in the PCI Express capability structure.

4.1.4 Hot-Plug

Each root port implements a hot-plug controller that performs the following:

- Messages to turn on / off / blink LEDs
- Presence and attention button detection
- Interrupt generation

The root port only allows hot-plug with modules (for example, ExpressCard*). Edge-connector based hot-plug is not supported.

4.1.4.1 Presence Detection

When a module is plugged in and power is supplied, the physical layer detects the presence of the device, and the root port sets SLSTS.PDS (B0:D28:F0/F1/F2/F3:Offset 5Ah:bit 6) and SLSTS.PDC (B0:D28:F0/F1/F2/F3:Offset 6h:bit 3). If SLCTL.PDE (B0:D28:F0/F1/F2/F3:Offset 58h:bit 3) and SLCTL.HPE (B0:D28:F0/F1/F2/F3:Offset 58h:bit 5) are both set, the root port also generates an interrupt.

When a module is removed (via the physical layer detection), the root port clears SLSTS.PDS and sets SLSTS.PDC. If SLCTL.PDE and SLCTL.HPE are both set, the root port also generates an interrupt.

4.1.4.2 Message Generation

When system software writes to SLCTL.AIC (B0:D28:F0/F1/F2/F3:Offset 58h:bits 7:6) or SLCTL.PIC (B0:D28:F0/F1/F2/F3:Offset 58h:bits 9:8), the root port sends a message down the link to change the state of LEDs on the module.

Writes to these fields are non-postable cycles, and the resulting message is a postable cycle. When receiving one of these writes, the root port performs the following:

- Changes the state in the register.
- Generates a completion into the upstream queue.
- Formulates a message for the downstream port if the field is written to regardless of if the field changed.
- Generates the message on the downstream port.

- When the last message of a command is transmitted, sets SLSTS.CCE (D28:F0/F1/F2/F3:Offset 58h:bit 4) to indicate the command has completed. If SLCTL.CCE and SLCTL.HPE (D28:F0/F1/F2/F3:Offset 58h:bit 5) are set, the root port generates an interrupt.

The command completed register (SLSTS.CC) applies only to commands issued by software to control the Attention Indicator (SLCTL.AIC), Power Indicator (SLCTL.PIC), or Power Controller (SLCTL.PCC). However, writes to other parts of the Slot Control Register would invariably end up writing to the indicators, power controller fields; Hence, any write to the Slot Control Register is considered a command and if enabled, results in a command complete interrupt. The only exception is a write to disable the command complete interrupt which does not result in a command complete interrupt.

A single write to the Slot Control register is considered as a single command, and hence receives a single command complete, even if the write affects more than one field in the Slot Control Register.

4.1.4.3 Attention Button Detection

When an attached device is ejected, the user can press an attention button. This results in a the PCI Express message "Attention_Button_Pressed" from the device. Upon receiving this message, the root port will set SLSTS.ABP (B0:D28:F0/F1/F2/F3:Offset 5Ah:bit 0).

If SLCTL.ABE (B0:D28:F0/F1/F2/F3:Offset 58h:bit 0) and SLCTL.HPE (B0:D28:F0/F1/F2/F3:Offset 58h:bit 5) are set, the hot-plug controller will also generate an interrupt. The interrupt is generated on an edge-event. If SLSTS.ABP is already set, a new interrupt is not generated.

4.1.4.4 SMI/SCI Generation

Interrupts for hot-plug events are not supported on legacy operating systems. To support hot-plug on non-PCI Express* aware operating systems, hot-plug events can be routed to generate SCI. To generate SCI, MPC.HPCE (B0:D28:F0/F1/F2/F3:Offset D8h:bit 30) must be set. When set, enabled hot-plug events will cause SMSCS.HPCS (B0:D28:F0/F1/F2/F3:Offset DCh:bit 30) to be set.

Additionally, BIOS workarounds for hot-plug can be supported by setting MPC.HPME (B0:D28:F0/F1/F2/F3:Offset D8h:bit 1). When this bit is set, hot-plug events can cause SMI status bits in SMSCS to be set. Supported hot-plug events and their corresponding SMSCS bit are:

- Command Completed - SCSCS.HPCCM (B0:D28:F0/F1/F2/F3:Offset DCh:bit 3)
- Presence Detect Changed - SMSCS.HPPDM (B0:D28:F0/F1/F2/F3:Offset DCh:bit 1)
- Attention Button Pressed - SMSCS.HPABM (B0:D28:F0/F1/F2/F3:Offset DCh:bit 2)
- Link Active State Changed - SMSCS.HPLAS (B0:D28:F0/F1/F2/F3:Offset DCh:bit 4)

When any of these bits are set, SMI # is generated. These bits are set regardless of whether interrupts or SCI is enabled for hot-plug events. The SMI# may occur concurrently with an interrupt or SCI.

4.2 LPC Bridge (with System and Management Functions) (B0:D31:F0)

The LPC bridge function resides in PCI Bus 0:Device 31:Function 0. In addition to the LPC bridge function, B0:D31:F0 contains other functional units including:

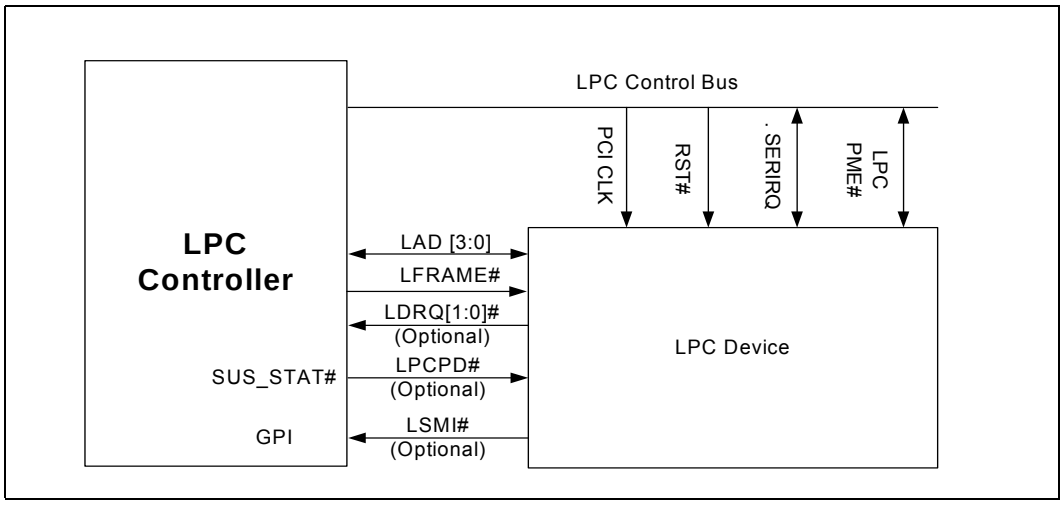
- DMA
- Interrupt controllers
- Timers
- Power Management
- System Management
- GPIO
- UART
- WDT
- RTC

In this chapter, registers and functions associated with other functional units (power management, GPIO, USB*, etc.) are described in their respective sections.

4.2.1 LPC Interface

The LPC interface as described in the *Low Pin Count Interface Specification,* Revision 1.1. The LPC interface is shown in Figure 4-1. The LPC interface implements all of the signals that are shown as optional, but peripherals are not required to do so.

Figure 4-1. LPC Interface Diagram



4.2.1.1 LPC Cycle Types

The LPC controller implements all of the cycle types described in the *Low Pin Count Interface Specification,* Revision 1.1. Table 4-2 shows the cycle types supported by the LPC controller.

Table 4-2. LPC Cycle Types Supported

Cycle Type	Comment
Memory Read	1 byte only. (See Note 1 below)
Memory Write	1 byte only. (See Note 1 below)
I/O Read	1 byte only. LPC Controller breaks up 16- and 32-bit processor cycles into multiple 8-bit transfers.
I/O Write	1 byte only. LPC Controller breaks up 16- and 32-bit processor cycles into multiple 8-bit transfers.
DMA Read	Can be 1, or 2 bytes
DMA Write	Can be 1, or 2 bytes
Bus Master Read	Can be 1, 2, or 4 bytes. (See Note 2 below)
Bus Master Write	Can be 1, 2, or 4 bytes. (See Note 2 below)

Notes:

1. The LPC Controller provides a single generic memory range (LGMR) for decoding memory cycles and forwarding them as LPC Memory cycles on the LPC bus. The LGMR memory decode range is 64 KB in size and can be defined as being anywhere in the 4 GB memory space. This range needs to be configured by BIOS during POST to provide the necessary memory resources. BIOS should advertise the LPC Generic Memory Range as Reserved to the OS in order to avoid resource conflict. For larger transfers, The LPC Controller performs multiple 8-bit transfers. If the cycle is not claimed by any peripheral, it is subsequently aborted, and the LPC Controller returns a value of all 1s to the processor. This is done to maintain compatibility with ISA memory cycles where pull-up resistors would keep the bus high if no device responds.
2. Bus Master Read or Write cycles must be naturally aligned. For example, a 1-byte transfer can be to any address. However, the 2-byte transfer must be word-aligned (for example, with an address where A0=0). A dword transfer must be dword-aligned (for example, with an address where A1 and A0 are both 0).

4.2.1.2 Start Field Definition

Table 4-3. Start Field Bit Definitions

Bits[3:0] Encoding	Definition
0000	Start of cycle for a generic target
0010	Grant for bus master 0
0011	Grant for bus master 1
1111	Stop/Abort: End of a cycle for a target.

Note: All other encodings are RESERVED.

4.2.1.3 Cycle Type / Direction (CYCTYPE + DIR)

The LPC controller always drives bit 0 of this field to 0. Peripherals running bus master cycles must also drive bit 0 to 0. Table 4-4 shows the valid bit encodings.

Table 4-4. Cycle Type Bit Definitions

Bits[3:2]	Bit1	Definition
00	0	I/O Read
00	1	I/O Write
01	0	Memory Read
01	1	Memory Write
10	0	DMA Read
10	1	DMA Write
11	x	Reserved. If a peripheral performing a bus master cycle generates this value, The LPC Controller aborts the cycle.

4.2.1.4 Size

Bits[3:2] are reserved. The LPC controller always drives them to 00. Peripherals running bus master cycles are supposed to drive 00 for bits 3:2, but the LPC controller ignores those bits. Bits[1:0] are encoded as listed in Table 4-5.

Table 4-5. Transfer Size Bit Definition

Bits[1:0]	Size
00	8-bit transfer (1 byte)
01	16-bit transfer (2 bytes)
10	Reserved. The LPC Controller never drives this combination. If a peripheral running a bus master cycle drives this combination, the Controller may abort the transfer.
11	32-bit transfer (4 bytes)

4.2.1.5 SYNC

Valid SYNC field definitions are as follows.

Table 4-6. SYNC Bit Definition

Bits[3:0]	Indication
0000	**Ready:** SYNC achieved with no error. For DMA transfers, this also indicates DMA request deassertion and no more transfers desired for that channel.
0101	**Short Wait:** Part indicating wait-states. For bus master cycles, the controller does not use this encoding. Instead, the controller uses the Long Wait encoding (see next encoding below).
0110	**Long Wait:** Part indicating wait-states, and many wait-states is added. This encoding driven by the controller for bus master cycles, rather than the Short Wait (0101).
1001	**Ready More (Used only by peripheral for DMA cycle):** SYNC achieved with no error and more DMA transfers desired to continue after this transfer. This value is valid only on DMA transfers and is not allowed for any other type of cycle.
1010	**Error:** Sync achieved with error. This is generally used to replace the SERR# or IOCHK# signal on the PCI/ISA bus. It indicates that the data is to be transferred, but there is a serious error in this transfer. For DMA transfers, this not only indicates an error, but also indicates DMA request deassertion and no more transfers desired for that channel.

Note: All other combinations are RESERVED.

4.2.1.6 SYNC Time-Out

Several error cases can occur on the LPC interface. The controller responds as defined in section 4.2.1.9 of the *Low Pin Count Interface Specification*, Revision 1.1 to the stimuli described therein. There may be other peripheral failure conditions, but these are not handled by the controller.

4.2.1.7 SYNC Error Indication

The LPC controller responds as defined in section 4.2.1.10 of the *Low Pin Count Interface Specification,* Revision 1.1.

Upon recognizing the SYNC field indicating an error, the controller treats this as an SERR by reporting this into the Device 31 Error Reporting Logic.

4.2.1.8 LFRAME# Usage

The controller follows the usage of LFRAME# as defined in the *Low Pin Count Interface Specification,* Revision 1.1.

The controller performs an abort for the following cases (possible failure cases):

- The controller starts a Memory, I/O, or DMA cycle, but no device drives a valid SYNC after four consecutive clocks.
- The controller starts a Memory, I/O, or DMA cycle, and the peripheral drives an invalid SYNC pattern.
- A peripheral drives an illegal address when performing bus master cycles.
- A peripheral drives an invalid value.

4.2.1.9 I/O Cycles

For I/O cycles targeting registers specified in the controller's decode ranges, the controller performs I/O cycles as defined in the *Low Pin Count Interface Specification,* Revision 1.1. These are 8-bit transfers. If the processor attempts a 16-bit or 32-bit transfer, the controller breaks the cycle up into multiple 8-bit transfers to consecutive I/O addresses.

Note: If the cycle is not claimed by any peripheral (and subsequently aborted), the controller returns a value of all 1s (FFh) to the processor. This is to maintain compatibility with ISA I/O cycles where pull-up resistors would keep the bus high if no device responds.

4.2.1.10 Bus Master Cycles

The LPC interface supports Bus Master cycles and requests (using LDRQ#) as defined in the *Low Pin Count Interface Specification,* Revision 1.1. The interface has two LDRQ# inputs, and thus supports two separate bus master devices. It uses the associated START fields for Bus Master 0 (0010b) or Bus Master 1 (0011b).

Note: The LPC interface does not support LPC Bus Masters performing I/O cycles. LPC Bus Masters should only perform memory read or memory write cycles.

4.2.1.11 LPC Power Management

LPCPD# Protocol

Same timings as for SUS_STAT#. Upon driving SUS_STAT# low, LPC peripherals drive LDRQ# low or tri-state it. The interface shuts off the LDRQ# input buffers. After driving SUS_STAT# active, The LPC controller drives LFRAME# low, and tri-states (or drive low) LAD[3:0].

Note: The *Low Pin Count Interface Specification,* Revision 1.1 defines the LPCPD# protocol where there is at least 30 µs from LPCPD# assertion to LRST# assertion. This specification explicitly states that this protocol only applies to entry/exit of low power states which does not include asynchronous reset events. The controller asserts both SUS_STAT# (connects to LPCPD#) and PLTRST# (connects to LRST#) at the same time during a global reset. This is not inconsistent with the LPC LPCPD# protocol.

4.2.1.12 Configuration and Implications

LPC I/F Decoders

To allow the I/O cycles and memory mapped cycles to go to the LPC interface, the interface includes several decoders. During configuration, the controller must be programmed with the same decode ranges as the peripheral. The decoders are programmed via the Device 31:Function 0 configuration space.

Bus Master Device Mapping and START Fields

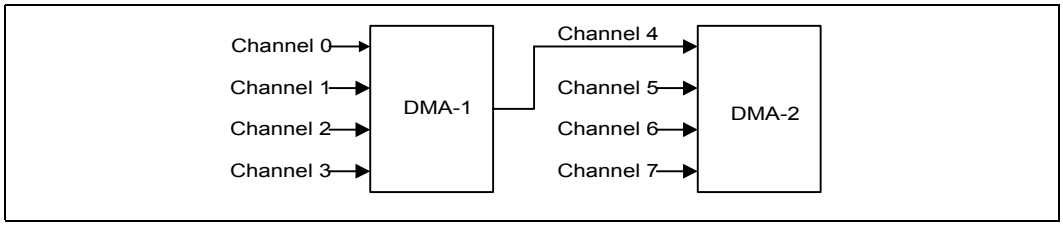
Bus Masters must have a unique START field. In the case where the LPC interface supports two LPC bus masters, it drives 0010 for the START field for grants to bus master #0 (requested via LDRQ0#) and 0011 for grants to bus master #1 (requested via LDRQ1#.). Thus, no registers are needed to configure the START fields for a particular bus master.

4.3 DMA Operation (B0:D31:F0)

LPC DMA operations are supported using an internal DMA controller. The DMA controller has registers that are fixed in the lower 64 KB of I/O space. The DMA controller is configured using registers in the PCI configuration space. These registers allow configuration of the channels for use by LPC DMA.

The DMA circuitry incorporates the functionality of two 82C37 DMA controllers with seven independently programmable channels (Figure 4-2). DMA controller 1 (DMA-1) corresponds to DMA channels 0–3 and DMA controller 2 (DMA-2) corresponds to channels 5–7. DMA channel 4 is used to cascade the two controllers and defaults to cascade mode in the DMA Channel Mode (DCM) Register. Channel 4 is not available for any other purpose. In addition to accepting requests from DMA slaves, the DMA controller also responds to requests that software initiates. Software may initiate a DMA service request by setting any bit in the DMA Channel Request Register to a 1.

Figure 4-2. DMA Controller



Each DMA channel is hardwired to the compatible settings for DMA device size: channels [3:0] are hardwired to 8-bit, count-by-bytes transfers, and channels [7:5] are hardwired to 16-bit, count-by-words (address shifted) transfers.

The DMA controller provides 24-bit addressing in compliance with the ISA-Compatible specification. Each channel includes a 16-bit ISA-Compatible Current Register which holds the 16 least-significant bits of the 24-bit address, an ISA-Compatible Page Register which contains the eight next most significant bits of address.

The DMA controller also features refresh address generation, and auto-initialization following a DMA termination.

4.3.1 Channel Priority

For priority resolution, the DMA consists of two logical channel groups: channels 0–3 and channels 4–7. Each group may be in either fixed or rotate mode, as determined by the DMA Command Register.

DMA I/O slaves normally assert their DREQ line to arbitrate for DMA service. However, a software request for DMA service can be presented through each channel's DMA Request Register. A software request is subject to the same prioritization as any hardware request.

4.3.1.1 Fixed Priority

The initial fixed priority structure is as follows:

High priority	Low priority
0, 1, 2, 3	5, 6, 7

The fixed priority ordering is 0, 1, 2, 3, 5, 6, and 7. In this scheme, channel 0 has the highest priority, and channel 7 has the lowest priority. Channels [3:0] of DMA-1 assume the priority position of channel 4 in DMA-2, taking priority over channels 5, 6, and 7.

4.3.1.2 Rotating Priority

Rotation allows for “fairness” in priority resolution. The priority chain rotates so that the last channel serviced is assigned the lowest priority in the channel group (0–3, 5–7).

Channels 0–3 rotate as a group of 4. They are always placed between channel 5 and channel 7 in the priority list.

Channel 5–7 rotate as part of a group of 4. That is, channels (5–7) form the first three positions in the rotation, while channel group (0–3) comprises the fourth position in the arbitration.

4.3.2 Address Compatibility Mode

When the DMA is operating, the addresses do not increment or decrement through the High and Low Page Registers. Therefore, if a 24-bit address is 01FFFFh and increments, the next address is 010000h, not 020000h. Similarly, if a 24-bit address is 020000h and decrements, the next address is 02FFFFh, not 01FFFFh.

When the DMA is operating in 16-bit mode, the addresses still do not increment or decrement through the High and Low Page Registers, but the page boundary is 128 K. Therefore, if a 24-bit address is 01FFFEh and increments, then the next address is 000000h, not 0100000h. Similarly, if a 24-bit address is 020000h and decrements, then the next address is 03FFFEh, not 02FFFEh. This is compatible with the 82C37 and Page Register implementation used in the PC-AT. This mode is set after CPURST is valid.

4.3.3 Summary of DMA Transfer Sizes

Table 4-7 lists the DMA device transfer sizes. "Current Byte/Word Count Register" indicates that the register contents represents either the number of bytes to transfer or the number of 16-bit words to transfer. "Current Address Increment/Decrement" indicates the number added to or taken from the Current Address register after each DMA transfer cycle. The DMA Channel Mode Register determines if the Current Address Register is incriminated or decremented.

4.3.3.1 Address Shifting When Programmed for 16-Bit I/O Count by Words

Table 4-7. DMA Transfer Size

DMA Device Date Size And Word Count	Current Byte/Word Count Register	Current Address Increment/Decrement
8-Bit I/O, Count By Bytes	Bytes	1
16-Bit I/O, Count By Words (Address Shifted)	Words	1

The DMA interface maintains compatibility with the implementation of the DMA in the PC AT that used the 82C37. The DMA shifts the addresses for transfers to/from a 16-bit device count-by-words.

Note: The least significant bit of the Low Page Register is dropped in 16-bit shifted mode. When programming the Current Address Register (when the DMA channel is in this mode), the Current Address must be programmed to an even address with the address value shifted right by one bit.

The address shifting is shown in Table 4-8.

Table 4-8. Address Shifting in 16-Bit I/O DMA Transfers

Output Address	8-Bit I/O Programmed Address (Ch 0–3)	16-Bit I/O Programmed Address (Ch 5–7) (Shifted)
A0 A[16:1] A[23:17]	A0 A[16:1] A[23:17]	0 A[15:0] A[23:17]

Note: The least significant bit of the Page Register is dropped in 16-bit shifted mode.

4.3.4 Autoinitialize

By programming a bit in the DMA Channel Mode Register, a channel may be set up as an autoinitialize channel. When a channel undergoes autoinitialization, the original values of the Current Page, Current Address and Current Byte/Word Count Registers are automatically restored from the Base Page, Address, and Byte/Word Count Registers of that channel following TC. The Base Registers are loaded simultaneously with the Current Registers by the microprocessor when the DMA channel is programmed and remain unchanged throughout the DMA service. The mask bit is not set when the channel is in autoinitialize. Following autoinitialize, the channel is ready to perform another DMA service, without processor intervention, as soon as a valid DREQ is detected.

4.3.5 Software Commands

There are three additional special software commands that the DMA controller can execute. The three software commands are:

- Clear Byte Pointer Flip-Flop
- Master Clear
- Clear Mask Register

They do not depend on any specific bit pattern on the data bus.

4.4 LPC DMA

DMA on LPC is handled through the use of the LDRQ# lines from peripherals and special encodings on LAD[3:0] from the host. Single, Demand, Verify, and Increment modes are supported on the LPC interface. Channels 0–3 are 8 bit channels. Channels 5–7 are 16-bit channels. Channel 4 is reserved as a generic bus master request.

4.4.1 Asserting DMA Requests

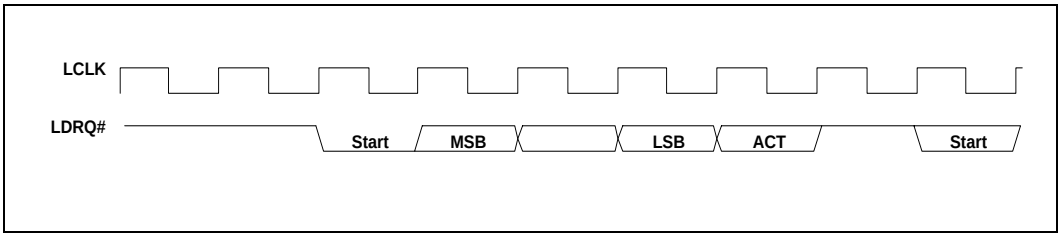
Peripherals that need DMA service encode their requested channel number on the LDRQ# signal. To simplify the protocol, each peripheral on the LPC I/F has its own dedicated LDRQ# signal (they may not be shared between two separate peripherals). The LPC interface has two LDRQ# inputs, allowing at least two devices to support DMA or bus mastering.

LDRQ# is synchronous with LCLK (PCI clock). As shown in Figure 4-3, the peripheral uses the following serial encoding sequence:

- Peripheral starts the sequence by asserting LDRQ# low (start bit). LDRQ# is high during idle conditions.
- The next three bits contain the encoded DMA channel number (MSB first).
- The next bit (ACT) indicates whether the request for the indicated DMA channel is active or inactive. The ACT bit is 1 (high) to indicate if it is active and 0 (low) if it is inactive. The case where ACT is low is rare, and is only used to indicate that a previous request for that channel is being abandoned.
- After the active/inactive indication, the LDRQ# signal must go high for at least 1 clock. After that one clock, LDRQ# signal can be brought low to the next encoding sequence.

If another DMA channel also needs to request a transfer, another sequence can be sent on LDRQ#. For example, if an encoded request is sent for channel 2, and then channel 3 needs a transfer before the cycle for channel 2 is run on the interface, the peripheral can send the encoded request for channel 3. This allows multiple DMA agents behind an I/O device to request use of the LPC interface, and the I/O device does not need to self-arbitrate before sending the message.

Figure 4-3. DMA Request Assertion through LDRQ#



4.4.2 Abandoning DMA Requests

DMA Requests can be deasserted in two fashions: on error conditions by sending an LDRQ# message with the 'ACT' bit set to 0, or normally through a SYNC field during the DMA transfer. This section describes boundary conditions where the DMA request needs to be removed prior to a data transfer.

There may be some special cases where the peripheral desires to abandon a DMA transfer. The most likely case of this occurring is due to a floppy disk controller which has overrun or underrun its FIFO, or software stopping a device prematurely.

In these cases, the peripheral wishes to stop further DMA activity. It may do so by sending an LDRQ# message with the ACT bit as 0. However, since the DMA request was seen by the controller, there is no guarantee that the cycle has not been granted and will shortly run on LPC. Therefore, peripherals must take into account that a DMA cycle may still occur. The peripheral can choose not to respond to this cycle, in which case the host will abort it, or it can choose to complete the cycle normally with any random data.

This method of DMA deassertion should be prevented whenever possible, to limit boundary conditions both on the controller and the peripheral.

4.4.3 General Flow of DMA Transfers

Arbitration for DMA channels is performed through the 8237 within the host. Once the host has won arbitration on behalf of a DMA channel assigned to LPC, it asserts LFRAME# on the LPC I/F and begins the DMA transfer. The general flow for a basic DMA transfer between the controller and an LPC peripheral is as follows:

1. The controller starts transfer by asserting 0000b on LAD[3:0] with LFRAME# asserted.
2. The controller asserts 'cycle type' of DMA, direction based on DMA transfer direction.
3. The controller asserts channel number and, if applicable, terminal count.
4. The controller indicates the size of the transfer: 8 or 16 bits.
5. If a DMA reads:
   - The controller drives the first 8 bits of data and turns the bus around.
   - The peripheral acknowledges the data with a valid SYNC.
   - If a 16-bit transfer, the process is repeated for the next 8 bits.
6. If a DMA writes:
   - The controller turns the bus around and waits for data.
   - The peripheral indicates data ready through SYNC and transfers the first byte.
   - If a 16-bit transfer, the peripheral indicates data ready and transfers the next byte.
7. The peripheral turns the bus around.

4.4.4 Terminal Count

Terminal count is communicated through LAD[3] on the same clock that DMA channel is communicated on LAD[2:0]. This field is the CHANNEL field. Terminal count indicates the last byte of transfer, based upon the size of the transfer.

For example, on an 8-bit transfer size (SIZE field is 00b), if the TC bit is set, then this is the last byte. On a 16-bit transfer (SIZE field is 01b), if the TC bit is set, then the second byte is the last byte. The peripheral, therefore, must internalize the TC bit when the CHANNEL field is communicated, and only signal TC when the last byte of that transfer size has been transferred.

4.4.5 Verify Mode

Verify mode is supported on the LPC interface. A verify transfer to the peripheral is similar to a DMA write, where the peripheral is transferring data to main memory. The indication from the host is the same as a DMA write, so the peripheral is driving data onto the LPC interface. However, the host does not transfer this data into main memory.

4.4.6 DMA Request Deassertion

An end of transfer is communicated to the controller through a special SYNC field transmitted by the peripheral. An LPC device must not attempt to signal the end of a transfer by deasserting LDREQ#. If a DMA transfer is several bytes (for example, a transfer from a demand mode device) the controller needs to know when to deassert the DMA request based on the data currently being transferred.

The DMA agent uses a SYNC encoding on each byte of data being transferred, which indicates to the controller whether this is the last byte of transfer or if more bytes are requested. To indicate the last byte of transfer, the peripheral uses a SYNC value of 0000b (ready with no error), or 1010b (ready with error). These encodings tell the controller that this is the last piece of data transferred on a DMA read (controller to peripheral), or the byte that follows is the last piece of data transferred on a DMA write (peripheral to controller).

When the controller sees one of these two encodings, it ends the DMA transfer after this byte and deasserts the DMA request to the 8237. Therefore, if the controller indicated a 16-bit transfer, the peripheral can end the transfer after one byte by indicating a SYNC value of 0000b or 1010b. The controller does not attempt to transfer the second byte, and deasserts the DMA request internally.

If the peripheral indicates a 0000b or 1010b SYNC pattern on the last byte of the indicated size, then the controller only deasserts the DMA request to the 8237 since it does not need to end the transfer.

If the peripheral wishes to keep the DMA request active, then it uses a SYNC value of 1001b (ready plus more data). This tells the 8237 that more data bytes are requested after the current byte has been transferred, so the controller keeps the DMA request active to the 8237. Therefore, on an 8-bit transfer size, if the peripheral indicates a SYNC value of 1001b to the controller, the data is transferred and the DMA request will remain active to the 8237. The controller will return lager with another START–CYCTYPE–CHANNEL–SIZE etc. combination to initiate another transfer to the peripheral.

The peripheral must not assume that the next START indication from the controller is another grant to the peripheral if it had indicated a SYNC value of 1001b. On a single mode DMA device, the 8237 will re-arbitrate after every transfer. Only demand mode DMA devices can be guaranteed that they will receive the next START indication from the controller.

Note: Indicating a 0000b or 1010b encoding on the SYNC field of an odd byte of a 16-bit channel (first byte of a 16-bit transfer) is an error condition.

- The host stops the transfer on the LPC bus as indicated, fills the upper byte with random data on DMA writes (peripheral to memory), and indicates to the 8237 that the DMA transfer occurred, incrementing the 8237's address and decrementing its byte count.

4.4.7 SYNC Field / LDRQ# Rules

Since DMA transfers on LPC are requested through an LDRQ# assertion message, and are ended through a SYNC field during the DMA transfer, the peripheral must obey the following rule when initiating back-to-back transfers from a DMA channel.

The peripheral must not assert another message for eight LCLKs after a deassertion is indicated through the SYNC field. This is needed to allow the 8237, that typically runs off a much slower internal clock, to see a message deasserted before it is re-asserted so that it can arbitrate to the next agent.

Under default operation, the host only performs 8-bit transfers on 8-bit channels and 16-bit transfers on 16-bit channels.

The method by which this communication between host and peripheral through system BIOS is performed is beyond the scope of this specification. Since the LPC host and LPC peripheral are motherboard devices, no "plug-n-play" registry is required.

The peripheral must not assume that the host is able to perform transfer sizes that are larger than the size allowed for the DMA channel, and be willing to accept a SIZE field that is smaller than what it may currently have buffered.

To that end, it is recommended that future devices that may appear on the LPC bus, that require higher bandwidth than 8-bit or 16-bit DMA allow, do so with a bus mastering interface and not rely on the 8237.

4.5 8254 Timers (B0:D31:F0)

There are three counters that have fixed uses. All registers and functions associated with the 8254 timers are in the core well. The 8254 unit is clocked by a 14.31818 MHz clock.

Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is typically programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value 1 counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two each counter period. The counter negates IRQ0 when the count value reaches 0. It then reloads the initial count value and again decrements the initial count value by two each counter period. The counter then asserts IRQ0 when the count value reaches 0, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

Counter 1, Refresh Request Signal

This counter provides the refresh request signal and is typically programmed for Mode 2 operation and only impacts the period of the REF_TOGGLE bit in Port 61. The initial count value is loaded one counter period after being written to the counter I/O address.

The REF_TOGGLE bit has a square wave behavior (alternate between 0 and 1) and will toggle at a rate based on the value in the counter. Programming the counter to anything other than Mode 2 results in undefined behavior for the REF_TOGGLE bit.

4.5.1 Timer Programming

The counter/timers are programmed as follows:

1. Write a control word to select a counter.
2. Write an initial count for the selected counter.
3. Load the least and/or most significant bytes (as required by Control Word bits 5, 4) of the 16-bit counter.
4. Repeat with other counters.

Two conventions must be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same counter. Otherwise, the counter is loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of all three counters. Several commands are available:

- **Control Word Command.** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command.** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command.** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

Table 4-9 lists the six operating modes for the interval counters.

Table 4-9. Counter Operating Modes

Mode	Function	Description
0	Out signal on end of count (=0)	Output is 0. When count goes to 0, output goes to 1 and stays at 1 until counter is reprogrammed.
1	Hardware retriggerable one-shot	Output is 0. When count goes to 0, output goes to 1 for one clock time.
2	Rate generator (divide by n counter)	Output is 1. Output goes to 0 for one clock time, then back to 1 and counter is reloaded.
3	Square wave output	Output is 1. Output goes to 0 when counter rolls over, and counter is reloaded. Output goes to 1 when counter rolls over, and counter is reloaded, etc.
4	Software triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.
5	Hardware triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.

4.5.2 Reading from the Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters: a simple read operation, counter Latch command, and the Read-Back command.

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

4.5.2.1 Simple Read

With a simple read operation, the counter is selected through port 40h (counter 0), 41h (counter 1), or 42h (counter 2).

Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of counter 2, the count can be stopped by writing to the GATE bit in port 61h.

4.5.2.2 Counter Latch Command

The Counter Latch command, written to port 43h, latches the count of a specific counter when the command is received. This command ensures that the count read from the counter is accurate, particularly when reading a two-byte count. The count value is then read from each counter’s Count register as was programmed by the Control register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

If a Counter is latched and then latched again before the count is read, the second Counter Latch command is ignored. The count read is the count when the first Counter Latch command was issued.

4.5.2.3 Read Back Command

The Read Back command, written to port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. If multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, returns the latched count. Subsequent reads return unlatched count.

4.6 8259 Interrupt Controllers (PIC) (B0:D31:F0)

The system incorporates the functionality of two 8259 interrupt controllers that provide interrupts for ISA compatible interrupts. These interrupts are: system timer, keyboard controller, serial ports, parallel ports, floppy disk, mouse, and DMA channels. In addition, this interrupt controller can support the PCI based interrupts, by mapping the PCI interrupt onto the compatible ISA interrupt line. Each 8259 core supports eight interrupts, numbered 0–7. Table 4-10 shows how the cores are connected.

Table 4-10. Interrupt Controller Core Connections

8259	8259 Input	Typical Interrupt Source	Connected Pin / Function
Master	0	Internal	Internal Timer / Counter 0 output / HPET #0
	1	Keyboard	IRQ1 via SERIRQ
	2	Internal	Slave controller INTR output
	3	Serial Port A	IRQ3 via SERIRQ, PIRQ#
	4	Serial Port B	IRQ4 via SERIRQ, PIRQ#
	5	Parallel Port / Generic	IRQ5 via SERIRQ, PIRQ#
	6	Floppy Disk	IRQ6 via SERIRQ, PIRQ#
	7	Parallel Port / Generic	IRQ7 via SERIRQ, PIRQ#
Slave	0	Internal Real Time Clock	Internal RTC / HPET #1
	1	Generic	IRQ9 via SERIRQ, SCI, TCO, or PIRQ#
	2	Generic	IRQ10 via SERIRQ, SCI, TCO, or PIRQ#
	3	Generic	IRQ11 via SERIRQ, SCI, TCO, or PIRQ#, or HPET #2
	4	PS/2 Mouse	IRQ12 via SERIRQ, SCI, TCO, or PIRQ#, or HPET #3
	5	Internal	State Machine output based on processor FERR# assertion. May optionally be used for SCI or TCO interrupt if FERR# not needed.
	6	SATA*	IRQ14 via SERIRQ or PIRQ#
	7	SATA*	IRQ15 via SERIRQ or PIRQ#

Note: SERIRQ and PIRQ are not externally routed. They are used to reference internally routed PCIe* interrupts.

The system cascades the slave controller onto the master controller through master controller interrupt input 2. This leaves 15 possible interrupts for the PIC. Interrupts can individually be programmed to be edge or level, except for IRQ0, IRQ2, IRQ8#, and IRQ13.

Active-low interrupt sources (for example, the PIRQ#s) are inverted internally. In the following descriptions of the 8259s, the interrupt levels are in reference to the signals at the internal interface of the 8259s after the required inversions have occurred. Therefore, the term "high" indicates "active," which means "low" on an originating PIRQ#.

4.6.1 Interrupt Handling

4.6.1.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 4-11 defines the IRR, ISR, and IMR.

Table 4-11. Interrupt Status Registers

Bit	Description
IRR	Interrupt Request Register - This bit is set on a low to high transition of the interrupt line in edge mode, and by an active high level in level mode. This bit is set whether or not the interrupt is masked. However, a masked interrupt does not generate INTR.
ISR	Interrupt Service Register - This bit is set, and the corresponding IRR bit cleared, when an interrupt acknowledge cycle is seen, and the vector returned is for that interrupt.
IMR	Interrupt Mask Register - This bit determines whether an interrupt is masked. Masked interrupts do not generate INTRs.

4.6.1.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle that is translated by the host bridge into a PCI Interrupt Acknowledge Cycle. The PIC translates this command into two internal INTA# pulses expected by the 8259 cores. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave sends the interrupt vector to the processor with the acknowledged interrupt code. This code is based upon bits [7:3] of the corresponding ICW2 register, combined with three bits representing the interrupt within that controller.

Table 4-12. Content of Interrupt Vector Byte

Master, Slave Interrupt	Bits [7:3]	Bits [2:0]
IRQ7,15	ICW2[7:3]	111
IRQ6,14		110
IRQ5,13		101
IRQ4,12		100
IRQ3,11		011
IRQ2,10		010
IRQ1,9		001
IRQ0,8		000

4.6.1.3 Hardware/Software Interrupt Sequence

1. One or more of the Interrupt Request lines (IRQ) are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor if an asserted interrupt is not masked.

3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle. The cycle is translated into a PCI interrupt acknowledge cycle by the host bridge. This command is broadcast over PCI.
4. Upon observing its own interrupt acknowledge cycle on PCI, the cycle is converted into two cycles that the internal 8259 pair can respond to. Each cycle appears as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.
5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine if it must respond with an interrupt vector during the second INTA# pulse.
6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. If no interrupt request is present because the request was too short in duration, the PIC returns vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

4.6.2 Initialization Command Words (ICWx)

Before the operation can begin, each 8259 must be initialized. This is a four-byte sequence. The four initialization command words are referred to by their acronyms: ICW1, ICW2, ICW3, and ICW4.

The base address for each 8259 initialization command word is a fixed location in the I/O memory space: 20h for the master controller and A0h for the slave controller.

4.6.2.1 ICW1

An I/O write to the master or slave controller base address with data bit 4 equal to 1 is interpreted as a write to ICW1. Upon sensing this write, the PIC expects three more byte writes to 21h for the master controller, or A1h for the slave controller, to complete the ICW sequence.

A write to ICW1 starts the initialization sequence during which the following automatically occur:

1. Following initialization, an interrupt request (IRQ) input must make a low-to-high transition to generate an interrupt.
2. The Interrupt Mask Register is cleared.
3. IRQ7 input is assigned priority 7.
4. The slave mode address is set to 7.
5. Special mask mode is cleared and Status Read is set to IRR.

4.6.2.2 ICW2

The second write in the sequence (ICW2) is programmed to provide bits [7:3] of the interrupt vector that is released during an interrupt acknowledge. A different base is selected for each interrupt controller.

4.6.2.3 ICW3

The third write in the sequence (ICW3) has a different meaning for each controller:

- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. In this case, IRQ2 is used. Therefore, bit 2 of ICW3 on the master controller is set to a 1, and the other bits are set to 0s.
- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller if the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and if it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

4.6.2.4 ICW4

The final write in the sequence (ICW4) must be programmed for both controllers. At the very least, bit 0 must be set to a 1 to indicate that the controllers are operating in an Intel Architecture-based system.

4.6.3 Operation Command Words (OCW)

These command words reprogram the Interrupt controller to operate in various interrupt modes.

- OCW1 masks and unmasks interrupt lines.
- OCW2 controls the rotation of interrupt priorities when in rotating priority mode, and controls the EOI function.
- OCW3 sets up ISR/IRR reads, enables/disables the special mask mode (SMM), and enables/disables polled interrupt mode.

4.6.4 Modes of Operation

4.6.4.1 Fully Nested Mode

In this mode, interrupt requests are ordered in priority from 0 through 7, with 0 being the highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the ISR for the interrupt is set. This ISR bit remains set until. The processor issues an EOI command immediately before returning from the service routine, or if in AEOI mode, on the trailing edge of the second INTA#. While the ISR bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels generate another interrupt. Interrupt priorities can be changed in the rotating priority mode.

4.6.4.2 Special Fully Nested Mode

This mode is used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully nested mode is programmed to the master controller. This mode is similar to the fully nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave are recognized by the master and initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.

- When exiting the Interrupt Service routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a Non-Specific EOI command to the slave and then reading its ISR. If it is 0, a non-specific EOI can also be sent to the master.

4.6.4.3 Automatic Rotation Mode (Equal Priority Devices)

In some applications, there are a number of interrupting devices of equal priority. Automatic rotation mode provides for a sequential 8-way rotation. In this mode, a device receives the lowest priority after being serviced. In the worst case, a device requesting an interrupt has to wait until each of seven other devices are serviced at most once.

There are two ways to accomplish automatic rotation using OCW2; the rotation on Non-Specific EOI Command (R=1, SL=0, EOI=1) and the rotate in automatic EOI mode that is set by (R=1, SL=0, EOI=0).

4.6.4.4 Specific Rotation Mode (Specific Priority)

Software can change interrupt priorities by programming the bottom priority. For example, if IRQ5 is programmed as the bottom priority device, then IRQ6 is the highest priority device. The Set Priority command is issued in OCW2 to accomplish this, where: R=1, SL=1, and LO–L2 is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes can be executed during an EOI command by using the Rotate on Specific EOI command in OCW2 (R=1, SL=1, EOI=1 and LO–L2=IRQ level to receive bottom priority.

4.6.4.5 Poll Mode

Poll mode can be used to conserve space in the interrupt vector table. Multiple interrupts that can be serviced by one interrupt service routine do not need separate vectors if the service routine uses the poll command. Poll mode can also be used to expand the number of interrupts. The polling interrupt service routine can call the appropriate service routine, instead of providing the interrupt vectors in the vector table. In this mode, the INTR output is not used and the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting P=1 in OCW3. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit if there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read contains a 1 in bit 7 if there is an interrupt, and the binary code of the highest priority level in bits 2:0.

4.6.4.6 Cascade Mode

The PIC has one master 8259 and one slave 8259 cascaded onto the master through IRQ2. This configuration can handle up to 15 separate priority levels. The master controls the slaves through a three bit internal bus. When the master drives 010b on this bus, the slave controller takes responsibility for returning the interrupt vector. An EOI command must be issued twice: once for the master and once for the slave.

4.6.4.7 Edge and Level Triggered Mode

In ISA systems this mode is programmed using bit 3 in ICW1, which sets level or edge for the entire controller. This bit is disabled and a new register for edge and level triggered mode selection, per interrupt input, is included. This is the Edge/Level Control Registers ELCR1 and ELCR2.

If an ELCR bit is 0, an interrupt request is recognized by a low-to-high transition on the corresponding IRQ input. The IRQ input can remain high without generating another interrupt. If an ELCR bit is 1, an interrupt request is recognized by a high level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

In both the edge and level triggered modes, the IRQ inputs must remain active until after the falling edge of the first internal INTA#. If the IRQ input goes inactive before this time, a default IRQ7 vector is returned.

4.6.4.8 End of Interrupt (EOI) Operations

An EOI can occur in one of two fashions: by a command word write issued to the PIC before returning from a service routine, the EOI command, or automatically when AEOI bit in ICW4 is set to 1.

4.6.4.9 Normal End of Interrupt

In normal EOI, software writes an EOI command before leaving the interrupt service routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific.

When a Non-Specific EOI command is issued, the PIC clears the highest ISR bit of those that are set to 1. Non-Specific EOI is the normal mode of operation of the PIC, as the interrupt being serviced currently is the interrupt entered with the interrupt acknowledge. When the PIC is operated in modes that preserve the fully nested structure, software can determine which ISR bit to clear by issuing a Specific EOI. An ISR bit that is masked is not cleared by a Non-Specific EOI if the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

4.6.4.10 Automatic End of Interrupt Mode

In this mode, the PIC automatically performs a Non-Specific EOI operation at the trailing edge of the last interrupt acknowledge pulse. From a system standpoint, this mode should be used only when a nested multi-level interrupt structure is not required within a single PIC. The AEOI mode can only be used in the master controller and not the slave controller.

4.6.5 Masking Interrupts

4.6.5.1 Masking on an Individual Interrupt Request

Each interrupt request can be masked individually by the Interrupt Mask Register (IMR). This register is programmed through OCW1. Each bit in the IMR masks one interrupt channel. Masking IRQ2 on the master controller masks all requests for service from the slave controller.

4.6.5.2 Special Mask Mode

Some applications may require an interrupt service routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The special mask mode enables all interrupts not masked by a bit set in the Mask register. Normally, when an interrupt service routine acknowledges an interrupt without issuing an EOI to clear the ISR bit, the interrupt controller inhibits all lower priority requests. In the special mask mode, any interrupts may be selectively enabled by loading the Mask Register with the appropriate pattern. The special mask mode is set by OCW3 where: SSMM=1, SMM=1, and cleared where SSMM=1, SMM=0.

4.6.6 Steering PCI Interrupts

The PIC can be programmed to allow PIRQA#-PIRQH# to be routed internally to interrupts 3–7, 9–12, 14 or 15. The assignment is programmable through the PIRQx Route Control registers, located at 60–63h and 68–6Bh in Device 31:Function 0. One or more PIRQx# lines can be routed to the same IRQx input. If interrupt steering is not required, the Route registers can be programmed to disable steering.

The PIRQx# lines are defined as active low, level sensitive to allow multiple interrupts on a PCI board to share a single line across the connector. When a PIRQx# is routed to specified IRQ line, software must change the IRQ's corresponding ELCR bit to level sensitive mode. Active-low interrupts (PIRQx#) can be internally inverted to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an active high device (through SERIRQ). However, active low interrupts can share their interrupt with PCI interrupts.

Internal sources of the PIRQs, including SCI and TCO interrupts, cause the external PIRQ to be asserted. The PIRQ input, like all of the other external sources, are routed accordingly.

4.7 Advanced Programmable Interrupt Controller (APIC) (B0:D31:F0)

In addition to the standard ISA-compatible PIC described in the previous chapter, the system incorporates the APIC. While the standard interrupt controller is intended for use in a uni-processor system, APIC can be used in either a uni-processor or multi-processor system.

4.7.1 Interrupt Handling

The I/O APIC handles interrupts differently than the 8259. Briefly, these differences are:

- Method of Interrupt Transmission - The I/O APIC transmits interrupts through memory writes on the normal data path to the processor, and interrupts are handled without the need for the processor to run an interrupt acknowledge cycle.
- Interrupt Priority - The priority of interrupts in the I/O APIC is independent of the interrupt number. For example, interrupt 10 can be given a higher priority than interrupt 3.
- More Interrupts - The I/O APIC supports a total of 24 interrupts.
- Multiple Interrupt Controllers - The I/O APIC architecture allows for multiple I/O APIC devices in the system with their own interrupt vectors.

4.7.2 Interrupt Mapping

The I/O APIC supports 24 APIC interrupts. Each interrupt has its own unique vector assigned by software. The interrupt vectors are mapped as follows and match "Config 6" of the *Multi-Processor Specification*.

Table 4-13. APIC Interrupt Mapping[1]

IRQ #	Via SERIRQ	Via PCI Message	Internal Modules
0	No	No	Cascade from 8259 #1
1	Yes	Yes	
2	No	No	8254 Counter 0, HPET #0 (legacy mode)
3	Yes	Yes	
4	Yes	Yes	
5	Yes	Yes	
6	Yes	Yes	
7	Yes	Yes	
8	No	No	RTC, HPET #1 (legacy mode)
9	Yes	Yes	Option for SCI, TCO
10	Yes	Yes	Option for SCI, TCO
11	Yes	Yes	HPET #2, Option for SCI, TCO (Note 2)
12	Yes	Yes	HPET #3 (Note 3)
13	No	No	FERR# logic
14	Yes	Yes	SATA*
15	Yes	Yes	SATA*
16	PIRQA#	Yes	Internal devices are routable
17	PIRQB#		
18	PIRQC#		
19	PIRQD#		
20	N/A	Yes	Option for SCI, TCO, HPET #0,1,2, 3. Other internal devices are routable
21	N/A		
22	N/A		
23	N/A		

Notes:
1. PIRQ is not externally routed rather they are used to reference internally routed PCIe* interrupts.
2. When programming the polarity of internal interrupt sources on the APIC, interrupts 0 through 15 receive active-high internal interrupt sources, while interrupts 16 through 23 receive active-low internal interrupt sources.
3. If IRQ 11 is used for HPET #2, software should ensure IRQ 11 is not shared with any other devices to guarantee the proper operation of HPET #2. The hardware does not prevent sharing of IRQ 11.
4. If IRQ 12 is used for HPET #3, software should ensure IRQ 12 is not shared with any other devices to guarantee the proper operation of HPET #3. The hardware does not prevent sharing of IRQ 12.

4.7.3 PCI Express* Message-Based Interrupts

When external devices through PCI Express* wish to generate an interrupt, they send the message defined in the *PCI Express* Base Specification,* Revision 1.0a for generating INTA# - INTD#. These are translated internal assertions/de-assertions of INTA# - INTD#.

4.7.4 IOxAPIC Address Remapping

To support Intel® Virtualization Technology, interrupt messages are required to go through similar address remapping as any other memory request. Address remapping allows for domain isolation for interrupts, so a device assigned in one domain is not allowed to generate an interrupt to another domain.

The address remapping is based on the Bus: Device: Function field associated with the requests. The internal APIC is required to initiate the interrupt message using a unique Bus: Device: function.

The system BIOS allows for the programming of the unique Bus: Device: Function address for the internal APIC. This address field does not change the APIC functionality and the APIC is not promoted as a stand-alone PCI device. See Bus 0: Device 31: Function 0 Offset 6Ch for additional information.

4.8 Serial Interrupt (B0:D31:F0)

Serial Interrupt protocol is supported. This allows a single signal to be used to report interrupt requests. The signal used to transmit this information is shared between the host and all peripherals that support serial interrupts. The signal line, SERIRQ, is synchronous to PCI clock, and follows the sustained tri-state protocol that is used by all PCI signals. This means that if a device has driven SERIRQ low, it first drives it high synchronous to PCI clock and release it the following PCI clock. The serial IRQ protocol defines this sustained tri-state signaling in the following fashion:

- S – Sample Phase. Signal driven low
- R – Recovery Phase. Signal driven high
- T – Turn-around Phase. Signal released

The SERIRQ interface supports a message for 21 serial interrupts. These represent the 15 ISA interrupts (IRQ0–1, 2–15), the four PCI interrupts, and the control signals SMI# and IOCHK#. The serial IRQ protocol does not support the additional APIC interrupts (20–23).

4.8.1 Start Frame

The serial IRQ protocol has two modes of operation which affect the start frame. These two modes are:

- Continuous - SERIRQ controller is responsible for generating the start frame.
- Quiet - a serial IRQ peripheral is responsible for beginning the start frame.

The mode that must first be entered when enabling the serial IRQ protocol is continuous mode. In this mode, SERIRQ Controller asserts the start frame. This start frame is 4, 6, or 8 PCI clocks wide based upon the Serial IRQ Control Register, bits 1:0 at 64h in Device 31:Function 0 configuration space. This is a polling mode.

When the serial IRQ stream enters quiet mode (signaled in the Stop Frame), the SERIRQ line remains inactive and pulled up between the Stop and Start Frame until a peripheral drives the SERIRQ signal low. The SERIRQ Controller senses the line low and continues to drive it low for the remainder of the Start Frame. Since the first PCI clock of the start frame was driven by the peripheral in this mode, the SERIRQ Controller drives the SERIRQ line low for 1 PCI clock less than in continuous mode. This mode of operation allows for a quiet, and therefore lower power, operation.

4.8.2 Data Frames

Once the Start frame has been initiated, all of the SERIRQ peripherals must start counting frames based on the rising edge of SERIRQ. Each of the IRQ/DATA frames has exactly 3 phases of 1 clock each:

- Sample Phase - During this phase, the SERIRQ device drives SERIRQ low if the corresponding interrupt signal is low. If the corresponding interrupt is high, then the SERIRQ devices tri-state the SERIRQ signal. The SERIRQ line remains high due to pull-up resistors (there is no internal pull-up resistor on this signal, an external pull-up resistor is required). A low level during the IRQ0–1 and IRQ2–15 frames indicates that an active-high ISA interrupt is not being requested, but a low level during the PCI INT[A:D], SMI#, and IOCHK# frame indicates that an active-low interrupt is being requested.
- Recovery Phase - During this phase, the device drives the SERIRQ line high if in the Sample Phase it was driven low. If it was not driven in the sample phase, it is tri-stated in this phase.
- Turn-around Phase - The device tri-states the SERIRQ line.

4.8.3 Stop Frame

After all data frames, a Stop Frame is driven by the SERIRQ controller. The SERIRQ signal is driven low by the SERIRQ controller for two or three PCI clocks. The number of clocks is determined by the SERIRQ configuration register. The number of clocks determines the next mode:

Table 4-14. Stop Frame Explanation

Stop Frame Width	Next Mode
2 PCI clocks	**Quiet Mode.** Any SERIRQ device may initiate a Start Frame
3 PCI clocks	**Continuous Mode.** Only the host (SERIRQ Controller) may initiate a Start Frame

4.8.4 Specific Interrupts Not Supported via SERIRQ

There are three interrupts seen through the serial stream that are not supported. These interrupts are generated internally, and are not sharable with other devices within the system. These interrupts are:

- IRQ0 - Heartbeat interrupt generated off of the internal 8254 counter 0.
- IRQ8# - RTC interrupt can only be generated internally.
- IRQ13 - Floating point error interrupt generated off of the processor assertion of FERR#.

The SERIRQ controller ignores the state of these interrupts in the serial stream, and does not adjust their level based on the level seen in the serial stream.

4.8.5 Data Frame Format

Table 4-15 shows the format of the data frames.

Table 4-15. Data Frame Format

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
1	IRQ0	2	Ignored. IRQ0 can only be generated via the internal 8524
2	IRQ1	5	
3	SMI#	8	Causes SMI# if low. Sets the SERIRQ_SMI_STS bit.
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	
8	IRQ7	23	
9	IRQ8	26	Ignored. IRQ8# can only be generated internally.
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	
14	IRQ13	41	Ignored. IRQ13 can only be generated from FERR#
15	IRQ14	44	Not attached to SATA* logic
16	IRQ15	47	Not attached to SATA* logic
17	IOCHCK#	50	Same as ISA IOCHCK# going active.
18	PCI INTA#	53	PIRQA#
19	PCI INTB#	56	PIRQB#
20	PCI INTC#	59	PIRQC#
21	PCI INTD#	62	PIRQD#

4.9 Real Time Clock (B0:D31:F0)

The Real Time Clock (RTC) module provides a battery backed-up date and time keeping device with two banks of static RAM with 128 bytes each, although the first bank has 114 bytes for general purpose usage.

Three interrupt features are available:

- Time of day alarm with once a second to once a month range
- Periodic rates of 122 µs to 500 ms
- End of update cycle notification

Seconds, minutes, hours, days, day of week, month, and year are counted. Daylight savings compensation is no longer supported. The hour is represented in 12- or 24-hour format, and data can be represented in BCD or binary format. The design is functionally compatible with the Motorola* MS146818B. The time keeping comes from a 32.768 kHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block have specific functions. The first 10 are for time and date information; the next four (0Ah to 0Dh) are registers, which configure and report RTC functions.

The time and calendar data should match the data mode (BCD or binary) and hour mode (12- or 24-hour) as selected in register B. The programmer must make sure that the data stored in these locations is within the reasonable values ranges and represents a possible date and time.

The exception to these ranges is to store a value of C0–FFh in the Alarm bytes to indicate a don't care situation. All Alarm conditions must match to trigger an Alarm Flag, which could trigger an Alarm Interrupt if enabled. The SET bit must be 1 while programming these locations to avoid clashes with an update cycle. Access to time and date information is done through the RAM locations. If a RAM read from the ten time and date bytes is attempted during an update cycle, the value read do not necessarily represent the true contents of those locations. Any RAM writes under the same conditions are ignored.

Note: The leap year determination for adding a 29th day to February does not take into account the end-of-the-century exceptions. The logic assumes that all years divisible by 4 are leap years. According to the Royal Observatory Greenwich, years that are divisible by 100 are typically not leap years. In every fourth century (years divisible by 400, like 2000), the 100-year-exception is over-ridden and a leap-year occurs. The year 2100 is the first time in which the current RTC implementation would incorrectly calculate the leap-year. The month/year alarms are not implemented.

4.9.1 Update Cycles

An update cycle occurs once a second, if the SET bit of register B is not asserted and the divide chain is properly configured. During this procedure, the stored time and date are incremented, overflow is checked, a matching alarm condition is checked, and the time and date are rewritten to the RAM locations. The update cycle starts at least 488 µs after the UIP bit of register A is asserted, and the entire cycle does not take more than 1984 µs to complete. The time and date RAM locations (0–9) are disconnected from the external bus during this time.

To avoid update and data corruption conditions, external RAM access to these locations can safely occur at two times. When a updated-ended interrupt is detected, almost 999 ms is available to read and write the valid time and date data. If the UIP bit of Register A is detected to be low, there is at least 488 µs before the update cycle begins.

Warning: The overflow conditions for leap years adjustments are based on more than one date or time item. To ensure proper operation when adjusting the time, the new time and data values should be set at least two seconds before leap year occurs.

4.9.2 Interrupts

The real-time clock interrupt is internally routed both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt is not shared with any other interrupt. IRQ8# from the SERIRQ stream is ignored. However, the High Performance Event Timers can also be mapped to IRQ8#; in this case, the RTC interrupt is blocked.

4.9.3 Lockable RAM Ranges

The RTC's battery-backed RAM supports two 8-byte ranges that can be locked via the configuration space. If the locking bits are set, the corresponding range in the RAM is not readable or writable. A write cycle to those locations has no effect. A read cycle to those locations does not return the location's actual value (resultant value is undefined).

Once a range is locked, the range can be unlocked only by a hard reset, which invokes the BIOS and allow it to relock the RAM range.

4.9.4 Century Rollover

A century rollover is detected when the Year byte (RTC I/O space, index offset 09h) transitions form 99 to 00. Upon detecting the rollover, the NEWCENTURY_STS bit (TCOBASE + 04h, bit 7). If the system is in an S0 state, this causes an SMI#. The SMI# handler can update registers in the RTC RAM that are associated with century value. If the system is in a sleep state (S1–S5) when the century rollover occurs, the NEWCENTURY_STS bit is set, but no SMI# is generated. When the system resumes from the sleep state, BIOS should check the NEWCENTURY_STS bit and update the century value in the RTC RAM.

4.9.5 Clearing Battery-Backed RTC RAM

CMOS RAM is cleared by using a jumper on RTCRST# or GPI. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low.

Using RTCRST# to Clear CMOS

A jumper on RTCRST# can be used to clear CMOS values, as well as reset to default, the state of those configuration bits that reside in the RTC power well. When the RTCRST# is strapped to ground, the RTC_PWR_STS bit (B0:D31:F0:A4h bit 2) is set and those configuration bits in the RTC power well is set to their default state. BIOS can monitor the state of this bit, and manually clear the RTC CMOS array once the system is booted. The normal position would cause RTCRST# to be pulled up through a weak pull-up resistor. Table 4-16 shows which bits are set to their default state when RTCRST# is asserted. This RTCRST# jumper technique allows the jumper to be moved and then replaced—all while the system is powered off. Then, once booted, the RTC_PWR_STS can be detected in the set state.

Table 4-16. Configuration Bits Reset by RTCRST# Assertion (Sheet 1 of 2)

Bit Name	Register	Location	Bit(s)	Default State
Alarm Interrupt Enable (AIE)	Register B (General Configuration) (RTC_REGB)	I/O space (RTC Index + 0Bh)	5	X
Alarm Flag (AF)	Register C (Flag Register) (RTC_REGC)	I/O space (RTC Index + 0Ch)	5	X
SWSMI_RATE_SEL	General PM Configuration 3 Register GEN_PMCON_3	B0:D31:F0:A4h	7:6	0
SLP_S4# Minimum Assertion Width	General PM Configuration 3 Register GEN_PMCON_3	B0:D31:F0:A4h	5:4	0
SLP_S4# Assertion Stretch Enable	General PM Configuration 3 Register GEN_PMCON_3	B0:D31:F0:A4h	3	0
RTC Power Status (RTC_PWR_STS)	General PM Configuration 3 Register GEN_PMCON_3	B0:D31:F0:A4h	2	0
Power Failure (PWR_FLR)	General PM Configuration 3 Register (GEN_PMCON_3)	B0:D31:F0:A4h	1	0
AFTERG3_EN	General PM Configuration 3 Register GEN_PMCON_3	B0:D31:F0:A4h	0	0
Power Button Override Status (PRBTNOR_STS)	Power Management 1 Status Register (PM1_STS)	PMBase + 00h	11	0

Table 4-16. Configuration Bits Reset by RTCRST# Assertion (Sheet 2 of 2)

Bit Name	Register	Location	Bit(s)	Default State
RTC Event Enable (RTC_EN)	Power Management 1 Enable Register (PM1_EN)	PMBase + 02h	10	0
Sleep Type (SLP_TYP)	Power Management 1 Control (PM1_CNT)	PMBase + 04h	12:10	0
PME_EN	General Purpose Event 0 Enables Register (GPE0_EN)	PMBase + 2Ch	11	0
BATLOW_EN	General Purpose Event 0 Enables Register (GPE0_EN)	PMBase + 2Ch	10	0
RI_EN	General Purpose Event 0 Enables Register (GPE0_EN)	PMBase + 2Ch	8	0
NEWCENTURY_STS	TCO1 Status Register (TCO1_STS)	TCOBase + 04h	7	0
Intruder Detect (INTRD_DET)	TCO2 Status Register (TCO2_STS)	TCOBase + 06h	0	0
Top Swap (TS)	Backed Up Control Register (BUC)	Chipset Config Registers:Offset 3414h	0	X

Using a GPI to Clear CMOS

A jumper on a GPI can also be used to clear CMOS values. BIOS would detect the setting of this GPI on system boot-up, and manually clear the CMOS array.

Note: The GPI strap technique to clear CMOS requires multiple steps to implement. The system is booted with the jumper in new position, then powered back down. The jumper is replaced back to the normal position, then the system is rebooted again.

Warning: Do not implement a jumper on VccRTC to clear CMOS.

4.10 Processor Interface (B0:D31:F0)

The processor interface is based on the following pin-based signals other than DMI:

- Standard Outputs to processor: A20GATE, INIT3_3V#, CPUPWRGD, PMSYNC#, PECI
- Standard Input from processor: THRMTRIP#

Most outputs to the processor use standard buffers. The system has separate V_CPU_IO signals that are pulled up at the system level to the processor voltage, and thus determines VOH for the outputs to the processor.

The following processor interface legacy pins were removed:

- IGNNE#, STPCLK#, DPSLP#, are DPRSLPVR are no longer required for systems.
- A20M#, SMI#, NMI, INIT#, INTR, FERR#: Functionality has been replaced by in-band Virtual Legacy Wire (VLW) messages. See Section 4.10.3.

4.10.1 Processor Interface Signals and VLW Messages

This section describes each of the processor interface signals. The behavior of some signals may vary during processor reset because the signals are used for frequency strapping.

4.10.1.1 A20M# (Mask A20) / A20GATE

The A20M# VLW message is asserted when both of the following conditions are true:

- The ALT_A20_GATE bit (Bit 1 of PORT92 register) is a 0
- The A20GATE input signal is a 0

The A20GATE input signal is expected to be generated by the external microcontroller (KBC).

4.10.1.2 INIT (Initialization)

The INIT# VLW Message is asserted based on any one of several events described in Table 4-17. When any of these events occur, INIT# is asserted for 16 PCI clocks, then driven high.

Table 4-17. INIT# Going Active

Cause of INIT3_3V# Going Active	Comment
Shutdown special cycle from processor observed on the Processor interconnect.	INIT assertion based on value of Shutdown Policy Select register (SPS)
PORT92 write, where INIT_NOW (bit 0) transitions from a 0 to a 1.	
PORTCF9 write, where SYS_RST (bit 1) was a 0 and RST_CPU (bit 2) transitions from 0 to 1.	
RCIN# input signal goes low. RCIN# is expected to be driven by the external microcontroller (KBC).	0 to 1 transition on RCIN# must occur before INIT3_3V# can be re-generated. ***Note:*** RCIN# signal is expected to be low during S3, S4, and S5 states. Transition on the RCIN# signal in those states (or the transition to those states) may not necessarily cause the INIT3_3V# signal to be generated to the processor.

4.10.1.3 FERR# (Numeric Coprocessor Error)

The system supports coprocessor error function with the FERR# message. The function is enabled via the COPROC_ERR_EN bit. If FERR# is driven active by the processor, IRQ13 goes active (internally). When it detects a write to the COPROC_ERR register (I/O Register F0h), the internal IRQ13 is negated and IGNNE# is activated. IGNNE# remains active until FERR# is driven inactive. IGNNE# is never driven active unless FERR# is active.

Note: IGNNE# (Ignore Numeric Error is internally generated by the processor.)

4.10.1.4 NMI (Non-Maskable Interrupt)

Non-Maskable Interrupts (NMIs) can be generated by several sources, as described in Table 4-18.

Table 4-18. NMI Sources

Cause of NMI	Comment
SERR# goes active (either internally, externally via SERR# signal, or via message from Processor)	Can instead be routed to generate an SCI, through the NMI2SCI_EN bit (Device 31:Function 0, TCO Base + 08h, bit 11).
IOCHK# goes active via SERIRQ# stream (ISA system Error)	Can instead be routed to generate an SCI, through the NMI2SCI_EN bit (Device 31:Function 0, TCO Base + 08h, bit 11).

4.10.1.5 Processor Power Good (CPUPWRGD)

This signal is connected to the processor's PWRGOOD input. CPUPWRGD is asserted after PCH PWROK. CPUPWRGD will not be asserted after PWROK until an internal DMI reference clock PLL is locked.

4.10.2 Dual-Processor Issues

4.10.2.1 Usage Differences

In dual-processor designs, some of the processor signals are unused or used differently than for uniprocessor designs.

- A20M#/A20GATE and FERR# are generally not used, but still supported.
- I/O APIC and SMI# are assumed to be used.

4.10.3 Virtual Legacy Wire (VLW) Messages

The system supports VLW messages as alternative method of conveying the status of the following legacy sideband interface signals to the CPU:

- A20M#
- INTR
- SMI#
- INIT#
- NMI

Note: IGNNE# VLW message is not required to be generated as it is internally emulated by the Processor.

VLW are inbound messages to the CPU. They are communicated via Vendor Defined Message over the DMI link.

Legacy processor signals can only be delivered via VLW. Delivery of legacy processor signals (A20M#, INTR, SMI#, INIT# or NMI) via I/O APIC controller is not supported.

4.11 Power Management (B0:D31:F0)

4.11.1 Features

- Support for *Advanced Configuration and Power Interface, Version 3.0b (ACPI)* providing power and thermal management

  - — ACPI 24-Bit Timer SCI and SMI# Generation
- PCI PME# signal for Wake Up from Low-Power states
- System Sleep State Control
  - — ACPI S3 state — Suspend to RAM (STR)
  - — ACPI S4 state — Suspend-to-Disk (STD)
  - — ACPI G2/S5 state — Soft Off (SOFF)
  - — Power Failure Detection and Recovery
- Management Engine Power Management Support
  - — Wake events from the Management Engine (enabled from all S-States including Catastrophic S5 conditions)

4.11.2 System Power States

Table 4-19 shows the power states defined for system platforms. The state names generally match the corresponding ACPI states.

Table 4-19. General Power States for Platform Systems

State/ Substates	Legacy Name / Description
G0/S0/C0	Full On: Processor operating. Individual devices may be shut down or be placed into lower power states to save power.
G0/S0/Cx	Cx State: Cx states are processor power states within the S0 system state that provide for various levels of power savings. The processor initiates C-state entry. The system's behavior is based on the processor state.
G1/S1	S1: The system provides the S1 messages and the S0 messages on a wake event. It is preferred for systems to use C-states than S1.
G1/S3	Suspend-To-RAM (STR): The system context is maintained in system DRAM, but power is shut off to non-critical circuits. Memory is retained and refreshes continue. All external clocks stop except RTC.
G1/S4	Suspend-To-Disk (STD): The context of the system is maintained on the disk. All power is then shut off to the system except for the logic required to resume.
G2/S5	Soft Off (SOFF): System context is not maintained. All power is shut off except for the logic required to restart. A full boot is required when waking.
G3	Mechanical OFF (MOFF): System context not maintained. All power is shut off except for the RTC. No "Wake" events are possible. This state occurs if the user removes the main system batteries in a mobile system, turns off a mechanical switch, or if the system power supply is at a level that is insufficient to power the "waking" logic. When system power returns, transition depends on the state just prior to the entry to G3 and the AFTERG3 bit in the GEN_PMCON3 register (B0:D31:F0, offset A4). See Table 4-26 for more details.

Table 4-20 shows the transitions rules among the various states. Transitions among the various states may appear to temporarily transition through intermediate states. For example, in going from S0 to S3, it may appear to pass through the G1/S1 states. These intermediate transitions and states are not listed in the table.

Table 4-20. State Transition Rules

Present State	Transition Trigger	Next State
G0/S0/C0	• DMI Msg • SLP_EN bit set • Power Button Override • Mechanical Off/Power Failure	• G0/S0/Cx • G1/Sx or G2/S5 state • G2/S5 • G3
G0/S0/Cx	• DMI Msg • Power Button Override • Mechanical Off/Power Failure	• G0/S0/C0 • S5 • G3
G1/S1, G1/S3, or G1/S4	• Any Enabled Wake Event • Power Button Override • Mechanical Off/Power Failure	• G0/S0/C0 (See Note 2) • G2/S5 • G3
G2/S5	• Any Enabled Wake Event • Mechanical Off/Power Failure	• G0/S0/C0 (See Note 2) • G3
G3	• Power Returns	• Optional to go to S0/C0 (reboot) or G2/S5 (stay off until power button pressed or other wake event). (See Notes 1 and 2)

Notes:
1. Some wake events can be preserved through power failure.
2. Transitions from the S1-S5 or G3 states to the S0 state are deferred until BATLOW# is inactive in mobile configurations.

4.11.3 System Power Planes

The system has several independent power planes, as described in Table 4-21. When a power plane is shut off, it should go to a 0 V level.

Table 4-21. System Power Plane

Plane	Controlled By	Description
CPU	SLP_S3# signal	The SLP_S3# signal can be used to cut the power to the processor completely.
MAIN	SLP_S3# signal	When SLP_S3# goes active, power can be shut off to any circuit not required to wake the system from the S3 state. Since the S3 state requires that the memory context be preserved, power must be retained to the main memory. The processor, devices on the PCI bus, LPC I/F, and graphics will typically be shut off when the Main power plane is off, although there may be small subsections powered.
MEMORY	SLP_S4# signal SLP_S5# signal	When SLP_S4# goes active, power can be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory can also be shut down. When SLP_S5# goes active, power can be shut off to any circuit not required to wake the system from the S5 state. Since the memory context does not need to be preserved in the S5 state, the power to the memory can also be shut.
DEVICE[n]	Implementation Specific	Individual subsystems may have their own power plane. For example, GPIO signals may be used to control the power to disk drives, audio amplifiers, or the display screen.
AUX	Implementation Specific	GbE MAC port 0 may have its own power plane for WOL implementation.
SUS	Implementation Specific	A sustained power plane used for S5 and RTC.

4.11.4 SMI#/SCI Generation

Upon any enabled SMI event taking place while the End of SMI (EOS) bit is set, the EOS bit is cleared and SMI is asserted to the processor, which causes it to enter SMM space. SMI assertion is performed using a Virtual Legacy Wire (VLW) message. Prior system generations (those based upon legacy processors) used an actual SMI# pin.

Once the SMI VLW has been delivered, no action is taken on behalf of active SMI events until Host software sets the End of SMI (EOS) bit. At that point, if any SMI events are still active, the system sends another SMI VLW message.

The SCI is a level-mode interrupt that is typically handled by an ACPI-aware operating system. In non-APIC systems (which is the default), the SCI IRQ is routed to one of the 8259 interrupts (IRQ 9, 10, or 11). The 8259 interrupt controller must be programmed to level mode for that interrupt.

In systems using the APIC, the SCI can be routed to interrupts 9, 10, 11, 20, 21, 22, or 23. The interrupt polarity changes depending on whether or not it is on an interrupt shareable with a PIRQ. The interrupt remains asserted until all SCI sources are removed.

Table 4-22 shows which events can cause an SMI and SCI. Some events can be programmed to cause either an SMI or SCI. The usage of the event for SCI (instead of SMI) is typically associated with an ACPI-based system. Each SMI or SCI source has a corresponding enable and status bit.

Table 4-22. Causes of SMI and SCI (Sheet 1 of 2)

Cause	SCI	SMI	Additional Enables	Where Reported
PME#	Yes	Yes	PME_EN=1	PME_STS
PME_B0 (Internal, Bus 0, PME-Capable Agents)	Yes	Yes	PME_B0_EN=1	PME_B0_STS
PCI Express* PME Messages	Yes	Yes	PCI_EXP_EN=1 (Not enabled for SMI)	PCI_EXP_STS
PCI Express* Hot Plug Message	Yes	Yes	HOT_PLUG_EN=1 (Not enabled for SMI)	HOT_PLUG_STS
Power Button Press	Yes	Yes	PWRBTN_EN=1	PWRBTN_STS
Power Button Override (Note 7)	Yes	No	None	PRBTNOR_STS
RTC Alarm	Yes	Yes	RTC_EN=1	RTC_STS
Ring Indicate	Yes	Yes	RI_EN=1	RI_STS
USB#1 wakes	Yes	Yes	USB1_EN=1	USB1_STS
USB#2 wakes	Yes	Yes	USB2_EN=1	USB2_STS
USB#3 wakes	Yes	Yes	USB3_EN=1	USB3_STS
USB#4 wakes	Yes	Yes	USB4_EN=1	USB4_STS
USB#5 wakes	Yes	Yes	USB5_EN=1	USB5_STS
USB#6 wakes	Yes	Yes	USB6_EN=1	USB6_STS
ACPI Timer overflow (2.34 sec.)	Yes	Yes	TMROF_EN=1	TMROF_STS
Any GPI[15:0]	Yes	Yes	GPI[x]_Route=10; GPI[x]_EN=1 (SCI) GPI[x]_Route=01; ALT_GPI_SMI[x]_EN=1 (SMI)	GPI[x]_STS ALT_GPI_SMI[x]_STS
GPIO[27]	Yes	Yes	GP27_EN=1	GP27_STS

Table 4-22. Causes of SMI and SCI (Sheet 2 of 2)

Cause	SCI	SMI	Additional Enables	Where Reported
TCO SCI Logic	Yes	No	TCOSCI_EN=1	TCOSCI_STS
TCO SCI message from CPU	Yes	No	none	CPUSCI_STS
TCO SMI Logic	No	Yes	TCO_EN=1	TCO_STS
TCO SMI — Year 2000 Rollover	No	Yes	none	NEWCENTURY_STS
TCO SMI — TCO TIMEROUT	No	Yes	none	TIMEOUT
TCO SMI — OS writes to TCO_DAT_IN register	No	Yes	none	SW_TCO_SMI
TCO SMI — Message from CPU	No	Yes	none	DMISMI_STS
TCO SMI — NMI occurred (and NMIs mapped to SMI)	No	Yes	NMI2SMI_EN=1	NMI2SMI_STS
TCO SMI — INTRUDER# signal goes active	No	Yes	INTRD_SEL=10	INTRD_DET
TCO SMI — Change of the BIOSWE (B0:D31:F0:DCh, bit 0) bit from 0 to 1	No	Yes	BLE=1	BIOSWR_STS
TCO SMI — Write attempted to BIOS	No	Yes	BIOSWE=1	BIOSWR_STS
BIOS_RLS written to	Yes	No	GBL_EN=1	GBL_STS
GBL_RLS written to	No	Yes	BIOS_EN=1	BIOS_STS
Write to B2h register	No	Yes	APMC_EN = 1	APM_STS
Periodic timer expires	No	Yes	PERIODIC_EN=1	PERIODIC_STS
64 ms timer expires	No	Yes	SWSMI_TMR_EN=1	SWSMI_TMR_STS
Enhanced USB* Legacy Support Event	No	Yes	LEGACY_USB2_EN = 1	LEGACY_USB2_STS
Enhanced USB* Intel Specific Event	No	Yes	INTEL_USB2_EN = 1	INTEL_USB2_STS
Serial IRQ SMI reported	No	Yes	none	SERIRQ_SMI_STS
Device monitors match address in its range	No	Yes	none	DEVTRAP_STS
SMBus Host Controller	No	Yes	SMB_SMI_EN Host Controller Enabled	SMBus host status reg.
SMBus Slave SMI message	No	Yes	none	SMBUS_SMI_STS
SMBus SMBALERT# signal active	No	Yes	none	SMBUS_SMI_STS
SMBus Host Notify message received	No	Yes	HOST_NOTIFY_INTREN	SMBUS_SMI_STS HOST_NOTIFY_STS
SLP_EN bit written to 1	No	Yes	SMI_ON_SLP_EN=1	SMI_ON_SLP_EN_STS
SPI Command Completed	No	Yes	None	SPI_SMI_STS
Software Generated GPE	Yes	Yes	SWGPE=1	SWGPE_STS
USB* Per-Port Registers Write Enable bit changes to 1	No	Yes	USB2_EN=1, Write_Enable_SMI_Enable =1	USB2_STS, Write Enable Status
GPIO Lockdown Enable bit changes from '1' to '0'	No	Yes	GPIO_UNLOCK_SMI_EN=1	GPIO_UNLOCK_SMI_STS

Notes:
1. SCI_EN must be 1 to enable SCI, except for BIOS_RLS. SCI_EN must be 0 to enable SMI.
2. SCI can be routed to cause interrupt 9:11 or 20:23 (20:23 only available in APIC mode).
3. GBL_SMI_EN must be 1 to enable SMI.
4. EOS must be written to 1 to re-enable SMI for the next 1.

5. The system must have SMI fully enabled when it is also enabled to trap cycles. If SMI is not enabled in conjunction with the trap enabling, then hardware behavior is undefined.
6. Only GPI[15:0] may generate an SMI or SCI.
7. When a power button override first occurs, the system transitions immediately to S5. The SCI only occurs after the next wake to S0 if the residual status bit (PRBTNOR_STS) is not cleared prior to setting SCI_EN.
8. GBL_STS being set causes an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place.

4.11.4.1 PCI Express* SCI

PCI Express ports and the processor (via DMI) have the ability to cause PME using messages. When a PME message is received, the PCI_EXP_STS bit is set. If the PCI_EXP_EN bit is also set, this causes an SCI via the GPE1_STS register.

4.11.4.2 PCI Express* Hot-Plug

PCI Express has a hot-plug mechanism and is capable of generating a SCI via the GPE1 register. It is also capable of generating an SMI. It is not capable of generating a wake event.

4.11.5 C-States

Systems implement C-states by having the processor control the states. Messages are exchanged with the processor as part of the C-state flow, but the system no longer directly controls any of the processor impacts of C-states, such as voltage levels or processor clocking. In addition to the new messages, the system also provides additional information to the processor using a sideband pin (PM_SYNC). The legacy C-state related pins (STPCLK#, STP_CPU#, DPRSLP#, DPRSLPVR#, etc.) do not exist.

4.11.6 Sleep States

4.11.6.1 Sleep State Overview

The system directly supports different sleep states (S1–S5), which are entered by methods such as setting the SLP_EN bit or due to a Power Button press. The entry to the Sleep states is based on several assumptions:

- The G3 state cannot be entered via any software mechanism.
- The G3 state indicates a complete loss of power.

4.11.6.2 Initiating Sleep State

Sleep states (S1–S5) are initiated by:

- Masking interrupts, turning off all bus master enable bits, setting the desired type in the SLP_TYP field, and then setting the SLP_EN bit. The hardware then attempts to gracefully put the system into the corresponding Sleep state.
- Pressing the PWRBTN# Signal for more than 4 seconds to cause a Power Button Override event. In this case the transition to the S5 state is less graceful, since there are no dependencies on DMI messages from the processor or on clocks other than the RTC clock.
- Assertion of the THRMTRIP# signal will cause a transition to the S5 state. This can occur when system is in S0 or S1 state.
- Internal watchdog timer timeout events

Table 4-23. Sleep Types

Sleep Type	Comment
S1	System lowers the processor's power consumption. No snooping is possible in this state.
S3	The system asserts SLP_S3#. The SLP_S3# signal controls the power to non-critical circuits. Power is only retained to devices needed to wake from this sleeping state, as well as to the memory.
S4	The system asserts SLP_S3# and SLP_S4#. The SLP_S4# signal shuts off the power to the memory subsystem. Only devices needed to wake from this state should be powered.
S5	The system asserts SLP_S3#, SLP_S4# and SLP_S5#.

4.11.6.3 Exiting Sleep States

Sleep states (S1–S5) are exited based on Wake events. The Wake events forces the system to a full on state (S0), although some non-critical subsystems might still be shut off and have to be brought back manually. For example, the hard disk may be shut off during a sleep state and have to be enabled via a GPIO pin before it can be used.

Upon exit from system-controlled Sleep states, the WAK_STS bit is set. The possible causes of Wake Events (and their restrictions) are shown in Table 4-24.

Table 4-24. Causes of Wake Events

Cause	States Can Wake From	How Enabled
RTC Alarm	S1–S5 (Note 1)	Set RTC_EN bit in PM1_EN register
Power Button	S1–S5	Always enabled as Wake event. (Note 2).
GPI[0:15]	S1–S5 (Note 1)	GPE0_EN register Note: GPIs that are in the core well are not capable of waking the system from sleep states when the core well is not powered.
GPIO[27]	S1-S5	Set GP27_EN in GPE0_EN Register
USB*	S1–S4	Set USB1_EN, USB 2_EN, USB3_EN, USB4_EN, USB5_EN, and USB6_EN bits in GPE0_EN register
RI#	S1–S5 (Note 1)	Set RI_EN bit in GPE0_EN register
Secondary PME#	S1–S5	Set PME_EN bit in GPE0_EN register.
PCI_EXP_WAKE#	S1–S5	PCI_EXP_WAKE bit (Note 3)
SATA*	S1	Set PME_EN bit in GPE0_EN register. (Note 4)
PCI_EXP PME Message	S1	Must use the PCI Express* WAKE# pin rather than messages for wake from S3,S4, or S5.
SMBALERT#	S1–S5	Always enabled as Wake event
SMBus Slave Wake Message (01h)	S1–S5	Wake/SMI# command always enabled as a Wake event. Note: SMBus Slave Message can wake the system from S1–S5, as well as from S5 due to Power Button Override. (Note 2).
SMBus Host Notify message received	S1–S5	HOST_NOTIFY_WKEN bit SMBus Slave Command register. Reported in the SMB_WAK_STS bit in the GPEO_STS register.
Intel® ME Non-Maskable Wake	S1-S5	Always enabled as a wake event. (Note 2).

Notes:

1. This is a wake event from S5 only if the sleep state was entered by setting the SLP_EN and SLP_TYP bits via software, or if there is a power failure.
2. If in the S5 state due to a power button override or THRMTRIP#, the possible wake events are due to Power Button, Wake SMBus Slave Message (01h), the ME-Initiated non-maskable Wake, the Integrated WOL Enable Override, Hard Reset Without Cycling (See Command Type 3 in Table 4-49), Hard Reset System (See Command Type 4 in Table 4-49).
3. When the WAKE# pin is active and the PCI Express* device is enabled to wake the system, the system will wake the platform.
4. SATA* can only trigger a wake event in S1, but if PME is asserted prior to S3/S4/S5 entry and software does not clear the PME_B0_STS, a wake event would still result.

The varous GPIs have different levels of functionality when used as wake events. The GPIs that reside in the core power well can only generate wake events from sleep states where the core well is powered. Only certain GPIs are ACPI-compliant, meaning that their Status and Enable bits reside in ACPI I/O space. Table 4-25 summarizes the use of GPIs as wake events.

Table 4-25. GPI Wake Events

GPI	Power Well	Wake From	Notes
GPI[7:0]	Core	S1	ACPI Compliant
GPI[15:8]	Suspend	S1–S5	ACPI Compliant

The latency to exit the various Sleep states varies and is dependent on power supply design to the extent that the exit latencies are insignificant.

4.11.6.4 PCI Express* WAKE# Signal and PME Event Message

PCI Express ports can wake the platform from any sleep state (S1, S3, S4, or S5) using the WAKE# pin. WAKE# is treated as a wake event, but does not cause any bits to go active in the GPE_STS register.

PCI Express ports and the processor (via DMI) have the ability to cause PME using messages. When a PME message is received, the PCI_EXP_STS bit is set.

4.11.6.5 Sx-G3-Sx, Handling Power Failures

Depending on when the power failure occurs and how the system is designed, different transitions could occur due to a power failure.

The AFTER_G3 bit provides the ability to program whether or not the system should boot once power returns after a power loss event. If the policy is to not boot, the system remains in an S5 state (unless previously in S4). Three possible events will wake the system after a power failure:

- PWRBTN#: PWRBTN# is always enabled as a wake event. When RSMRST# is low (G3 state), the PWRBTN_STS bit is reset. When the system exits G3 after power returns (RSMRST# goes high), the PWRBTN# signal is already high (because V_{CC}-standby goes high before RSMRST# goes high) and the PWRBTN_STS bit is 0.
- RI#: RI# does not have an internal pull-up. Therefore, if this signal is enabled as a wake event, it is important to keep this signal powered during the power loss event. If this signal goes low (active), when power returns the RI_STS bit is set and the system interprets that as a wake event.
- RTC Alarm: The RTC_EN bit is in the RTC well and is preserved after a power loss. Like PWRBTN_STS the RTC_STS bit is cleared when RSMRST# goes low.

The system monitors both PWROK and RSMRST# to detect for power failures. If PWROK goes low, the PWROK_FLR bit is set. If RSMRST# goes low, PWR_FLR is set.

Note: Although PME_EN is in the RTC well, this signal cannot wake the system after a power loss. PME_EN is cleared by RTCRST#, and PME_STS is cleared by RSMRST#.

Table 4-26. Transitions Due to Power Failure

State at Power Failure	AFTERG3_EN bit	Transition When Power Returns
S0, S1, S3	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0

4.11.7 Event Input Signals and Their Usage

There are various input signals that trigger specific events. This section describes those signals and how they should be used.

4.11.7.1 PWRBTN# (Power Button)

The PWRBTN# signal operates as a "Fixed Power Button" as described in the *Advanced Configuration and Power Interface, Version 2.0b.* PWRBTN# signal has a 16 ms de-bounce on the input. The state transition descriptions are included in Table 4-27. The transitions start as soon as the PWRBTN# is pressed (but after the debounce logic), and does not depend on when the power button is released.

Note: During the time that the SLP_S4# signal is stretched for the minimum assertion width (if enabled), the power button is not a wake event. See Power Button Override Function below.

Table 4-27. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI or SCI generated (depending on SCI_EN, PWRBTN_EN and GLB_SMI_EN)	Software typically initiates a Sleep state
S1–S5	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup
G3	PWRBTN# pressed	None	No effect since no power Not latched nor detected
S0–S4	PWRBTN# held low for at least 4 consecutive seconds	Unconditional transition to S5 state	No dependence on processor (DMI Messages) or any other subsystem

Power Button Override Function

If PWRBTN# is observed active for at least four consecutive seconds, the state machine should unconditionally transition to the G2/S5 state, regardless of present state (S0–S4), even if PWROK is not active. In this case, the transition to the G2/S5 state should not depend on any particular response from the processor (for example, a DMI Messages), nor any similar dependency from any other subsystem.

The PWRBTN# status is readable to check if the button is currently being pressed or has been released. The status is taken after the de-bounce, and is readable via the PWRBTN_LVL bit.

Note: The 4-second PWRBTN# assertion should only be used if a system lock-up has occurred. The 4-second timer starts counting when the system is in a S0 state. If the PWRBTN# signal is asserted and held active when the system is in a suspend state (S1–S5), the assertion causes a wake event. Once the system has resumed to the S0 state, the 4-second timer starts.

During the time that the SLP_S4# signal is stretched for the minimum assertion width (if enabled by B0:D31:F0:A4h bit 3), the Power Button is not a wake event. As a result, it is conceivable that the user will press and continue to hold the Power Button waiting for the system to awake. Since a 4-second press of the Power Button is already defined as an Unconditional Power down, the power button timer is forced to inactive while the power-cycle timer is in progress. Once the power-cycle timer has expired, the Power Button awakes the system. Once the minimum SLP_S4# power cycle expires, the Power Button must be pressed for another 4 to 5 seconds to create the Override condition to S5.

Sleep Button

The *Advanced Configuration and Power Interface, Version 2.0b* defines an optional Sleep button. It differs from the power button in that it only is a request to go from S0 to S1–S4 (not S5). In an S5 state, the Power Button can wake the system, but the Sleep Button cannot.

Although there is no specific signal designated as a Sleep Button, one of the GPIO signals can be used to create a "Control Method" Sleep Button. See the *Advanced Configuration and Power Interface, Version 2.0b* for implementation details.

4.11.7.2 RI# (Ring Indicator)

The Ring Indicator can cause a wake event (if enabled) from the S1–S5 states. Table 4-28 shows when the wake event is generated or ignored in different states. If in the G0/S0/Cx states, the system generates an interrupt based on RI# active, and the interrupt is set up as a Break event.

Table 4-28. Transitions Due to RI# Signal

Present State	Event	RI_EN	Event
S0	RI# Active	X	Ignored
S1–S5	RI# Active	0 1	Ignored Wake Event

Note: Filtering/Debounce on RI# is not implemented. If required, implement externally.

4.11.7.3 SYS_RESET# Signal

When the SYS_RESET# pin is detected as active after the 16 ms debounce logic, the system attempts to perform a "graceful" reset by waiting up to 25 ms for the SMBus to go idle. If the SMBus is idle when the pin is detected active, the reset occurs immediately; otherwise, the counter starts. If at any point during the count the SMBus goes idle the reset occurs. If, however, the counter expires and the SMBus is still active, a reset is forced upon the system even though activity is still occurring.

Once the reset is asserted, it remains asserted for 5 to 6 ms regardless of whether the SYS_RESET# input remains asserted or not. It cannot occur again until SYS_RESET# has been detected inactive after the debounce logic, and the system is back to a full S0 state with PLTRST# inactive. If bit 3 of the CF9h I/O register is set then SYS_RESET# results in a full power cycle reset.

4.11.7.4 THRMTRIP# Signal

If THRMTRIP# goes active, the processor is indicating an overheat condition, the system immediately transitions to an S5 state, driving SLP_S3#, SLP_S4#, SLP_S5# low, and setting the CTS bit. The transition looks like a power button override.

When a THRMTRIP# event occurs, the system will power down immediately without following the normal S0 -> S5 path. The system will immediately drive SLP_S3#, SLP_S4#, and SLP_S5# low after sampling THRMTRIP# active.

If the processor is running extremely hot and is heating up, it is possible (although unlikely) that components around it are no longer executing cycles properly. Therefore, if THRMTRIP# goes active, and the system is relying on state machine logic to perform the power down, the state machine may not be working, and the system does not power down.

The system provides filtering for short low glitches on the THRMTRIP# signal to prevent erroneous system shut downs from noise. Glitches shorter than 25 nsec are ignored.

During boot, THRMTRIP# is ignored until SLP_S3#, PWROK, and PLTRST# are all '1'. During entry into a powered-down state (due to S3, S4, S5 entry, power cycle reset, etc.) THRMTRIP# is ignored until either SLP_S3# = 0, or PWROK = 0, or SYS_PWROK = 0.

A thermal trip event will:

- Set the AFTERG3_EN bit
- Clear the PWRBTN_STS bit

- Clear all the GPE0_EN register bits
- Clear the SMB_WAK_STS bit only if SMB_SAK_STS was set due to SMBus slave receiving message and not set due to SMBAlert

4.11.8 ALT Access Mode

Before entering a low power state, several registers from powered down parts may need to be saved. In the majority of cases, this is not an issue, as registers have read and write paths. However, several of the ISA compatible registers are either read only or write only. To get data out of write-only registers, and to restore data into read-only registers, the system implements an ALT access mode.

If the ALT access mode is entered and exited after reading the registers of the timer (8254), the timer starts counting faster (13.5 ms). The following steps can cause problems:

1. BIOS enters ALT access mode for reading timer related registers.
2. BIOS exits ALT access mode.
3. BIOS continues through the execution of other needed steps and passes control to the operating system.

After getting control in step #3, if the operating system does not reprogram the system timer again, the timer ticks may be happening faster than expected. For example Microsoft* MS-DOS* and its associated software assume that the system timer is running at 54.6 ms and as a result the time-outs in the software may be happening faster than expected.

Operating systems (for example, Microsoft Windows* 98 and Windows* 2000) reprogram the system timer and therefore do not encounter this problem.

For other OS's (for example, Microsoft MS-DOS*) the BIOS should restore the timer back to 54.6 ms before passing control to the operating system. If the BIOS is entering ALT access mode before entering the suspend state it is not necessary to restore the timer contents after the exit from ALT access mode.

4.11.8.1 Write Only Registers with Read Paths in ALT Access Mode

The registers described in Table 4-29 have read paths in ALT access mode. The access number field in the table indicates which register is returned per access to that port.

Table 4-29. Write Only Registers with Read Paths in ALT Access Mode (Sheet 1 of 2)

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
00h	2	1	DMA Chan 0 base address low byte	40h	7	1	Timer Counter 0 status, bits [5:0]
		2	DMA Chan 0 base address high byte			2	Timer Counter 0 base count low byte
01h	2	1	DMA Chan 0 base count low byte			3	Timer Counter 0 base count high byte
		2	DMA Chan 0 base count high byte			4	Timer Counter 1 base count low byte
02h	2	1	DMA Chan 1 base address low byte			5	Timer Counter 1 base count high byte
		2	DMA Chan 1 base address high byte			6	Timer Counter 2 base count low byte
03h	2	1	DMA Chan 1 base count low byte			7	Timer Counter 2 base count high byte
		2	DMA Chan 1 base count high byte	41h	1		Timer Counter 1 status, bits [5:0]
04h	2	1	DMA Chan 2 base address low byte	42h	1		Timer Counter 2 status, bits [5:0]
		2	DMA Chan 2 base address high byte	70h	1		Bit 7 = NMI Enable, Bits [6:0] = RTC Address
05h	2	1	DMA Chan 2 base count low byte	C4h	2	1	DMA Chan 5 base address low byte
		2	DMA Chan 2 base count high byte			2	DMA Chan 5 base address high byte
06h	2	1	DMA Chan 3 base address low byte	C6h	2	1	DMA Chan 5 base count low byte
		2	DMA Chan 3 base address high byte			2	DMA Chan 5 base count high byte
07h	2	1	DMA Chan 3 base count low byte	C8h	2	1	DMA Chan 6 base address low byte
		2	DMA Chan 3 base count high byte			2	DMA Chan 6 base address high byte
08h	6	1	DMA Chan 0–3 Command[2]	CAh	2	1	DMA Chan 6 base count low byte
		2	DMA Chan 0–3 Request			2	DMA Chan 6 base count high byte
		3	DMA Chan 0 Mode: Bits(1:0) = 00	CCh	2	1	DMA Chan 7 base address low byte
		4	DMA Chan 1 Mode: Bits(1:0) = 01			2	DMA Chan 7 base address high byte
		5	DMA Chan 2 Mode: Bits(1:0) = 10	CEh	2	1	DMA Chan 7 base count low byte
		6	DMA Chan 3 Mode: Bits(1:0) = 11.			2	DMA Chan 7 base count high byte

Table 4-29. Write Only Registers with Read Paths in ALT Access Mode (Sheet 2 of 2)

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
20h	12	1	PIC ICW2 of Master controller	D0h	6	1	DMA Chan 4–7 Command[2]
		2	PIC ICW3 of Master controller			2	DMA Chan 4–7 Request
		3	PIC ICW4 of Master controller			3	DMA Chan 4 Mode: Bits(1:0) = 00
		4	PIC OCW1 of Master controller[1]			4	DMA Chan 5 Mode: Bits(1:0) = 01
		5	PIC OCW2 of Master controller			5	DMA Chan 6 Mode: Bits(1:0) = 10
		6	PIC OCW3 of Master controller			6	DMA Chan 7 Mode: Bits(1:0) = 11.
		7	PIC ICW2 of Slave controller				
		8	PIC ICW3 of Slave controller				
		9	PIC ICW4 of Slave controller				
		10	PIC OCW1 of Slave controller[1]				
		11	PIC OCW2 of Slave controller				
		12	PIC OCW3 of Slave controller				

Notes:
1. The OCW1 register must be read before entering ALT access mode.
2. Bits 5, 3, 1, and 0 return 0.

4.11.8.2 PIC Reserved Bits

Many bits within the PIC are reserved, and must have certain values written for the PIC to operate properly. Therefore, there is no need to return these values in ALT access mode. When reading PIC registers from 20h and A0h, the reserved bits return the values listed in Table 4-30.

Table 4-30. PIC Reserved Bits Return Values

PIC Reserved Bits	Value Returned
ICW2(2:0)	000
ICW4(7:5)	000
ICW4(3:2)	00
ICW4(0)	0
OCW2(4:3)	00
OCW3(7)	0
OCW3(5)	Reflects bit 6
OCW3(4:3)	01

4.11.8.3 Read Only Registers with Write Paths in ALT Access Mode

The registers described in Table 4-31 have write paths to them in ALT access mode. Software restores these values after returning from a powered down state. Software must handle these registers. When in normal mode, writing to the base address/count register also writes to the current address/count register. Therefore, the base address/count must be written first, then the part is put into ALT access mode and the current address/count register is written.

Table 4-31. Register Write Accesses in ALT Access Mode

I/O Address	Register Write Value
08h	DMA Status Register for channels 0–3.
D0h	DMA Status Register for channels 4–7.

4.11.9 System Power Supplies, Planes, and Signals

4.11.9.1 Power Plane Control with SLP_S3#, SLP_S4# and SLP_S5#

The SLP_S3# output signal can be used to cut power to the system core supply, since it only goes active for the Suspend-to-RAM state (typically mapped to ACPI S3). Power must be maintained to the suspend well, and to any other circuits that need to generate Wake signals from the Suspend-to-RAM state. During S3 (Suspend-to-RAM) all signals attached to powered down plans is tri-stated or driven low, unless they are pulled via a pull-up resistor.

Cutting power to the core may be done via the power supply, or by external FETs on the motherboard.

The SLP_S4# or SLP_S5# output signal can be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done via the power supply, or by external FETs on the motherboard.

The SLP_S4# output signal is used to remove power to additional subsystems that are powered during SLP_S3#.

SLP_S5# output signal can be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done via the power supply, or by external FETs on the motherboard.

4.11.9.2 SLP_S4# and Suspend-To-RAM Sequencing

The system memory suspend voltage regulator is controlled by the Glue logic. The SLP_S4# signal should be used to remove power to system memory rather than the SLP_S5# signal. The SLP_S4# logic provides a mechanism to fully cycle the power to the DRAM and/or detect if the power is not cycled for a minimum time.

Note: To utilize the minimum DRAM power-down feature that is enabled by the SLP_S4# Assertion Stretch Enable bit (B0:D31:F0:A4h bit 3), the DRAM power must be controlled by the SLP_S4# signal.

4.11.9.3 PWROK Signal

When asserted, PWROK is an indication that its core well power rails have been stable for at least 1 ms. PWROK can be driven asynchronously. When PWROK is low, the system asynchronously asserts PLTRST#. PWROK must not glitch, even if RSMRST# is low.

Power associated with PCI/PCIe* must be valid for 99 ms prior to PWROK assertion to comply with the 100 ms PCI 2.3 / PCIe* 1.1 specification on PLTRST# deassertion.

Note: SYS_RESET# is recommended for implementing the system reset button. This saves external logic that is needed if the PWROK input is used, and it allows for better handling of the SMBus and processor resets and avoids improperly reporting power failures.

4.11.9.4 BATLOW# (Battery Low)

The BATLOW# input can inhibit waking from S3, S4, and S5 states if power is not sufficient. It also causes an SMI if the system is already in an S0 state.

4.11.10 Legacy Power Management Theory of Operation

Instead of relying on ACPI software, legacy power management uses BIOS and various hardware mechanisms. The scheme relies on the concept of detecting when individual subsystems are idle, detecting when the whole system is idle, and detecting when accesses are attempted to idle subsystems.

However, the operating system is assumed to be at least APM enabled. Without APM calls, there is no quick way to know when the system is idle between keystrokes. The system does not support burst modes.

4.11.10.1 APM Power Management

There is a timer that, when enabled by the 1MIN_EN bit in the SMI Control and Enable register, generates an SMI once per minute. The SMI handler can check for system activity by reading the DEVTRAP_STS register. If none of the system bits are set, the SMI handler can increment a software counter. When the counter reaches a sufficient number of consecutive minutes with no activity, the SMI handler can then put the system into a lower power state.

If there is activity, various bits in the DEVTRAP_STS register is set. Software clears the bits by writing a 1 to the bit position.

The DEVTRAP_STS register allows for monitoring various internal devices, or Super I/O devices (SP, PP, FDC) on LPC or PCI, keyboard controller accesses, or audio functions on LPC or PCI. Other PCI activity can be monitored by checking the PCI interrupts.

4.11.11 Reset Behavior

When a reset is triggered, the system will send a warning message to the CPU to allow the CPU to attempt to complete any outstanding memory cycles and put memory into a safe state before the platform is reset. When the CPU is ready, it will send an acknowledge message to the system. Once the message is received the system asserts PLTRST#.

The system does not require an acknowledge message from the CPU to trigger PLTRST#. A global reset will occur after 4 seconds if an acknowledge from the CPU is not received.

When the system causes a reset by asserting PLTRST# its output signals will go to their reset states.

A reset in which the host platform is reset and PLTRST# is asserted is called a Host Reset or Host Partition Reset. Depending on the trigger a host reset may also result in power cycling. See the following table for details. If a host reset is triggered and the timer times out before receiving an acknowledge message from the CPU, a Global Reset with power cycle will occur. A reset in which the host and ME partitions of the platform are reset is called a Global Reset.

Table 4-32. Causes of Host and Global Resets

Trigger	Host Reset without Power Cycle	Host Reset with Power Cycle	Global Reset with Power Cycle
Write of 0Eh to CF9h Register when Global Reset Bit=0b	No	Yes	No (Note 1)
Write of 06h to CF9h Register when Global Reset Bit=0b	Yes	No	No (Note 1)
Write of 06h or 0Eh to CF9h Register when Global Reset Bit=1b	No	No	Yes
SYS_RESET# Asserted and CF9h Bit 3 = 0	Yes	No	No (Note 1)
SYS_RESET# Asserted and CF9h Bit 3 = 1	No	Yes	No (Note 1)
SMBus Slave Message received for Reset with Power Cycle	No	Yes	No (Note 1)
SMBus Slave Message received for Reset without Power Cycle	Yes	No	No (Note 1)
TCO Watchdog Timer reaches zero two times	Yes	No	No (Note 1)
Power Failure: PWROK signal or SYS_PWROK signal goes inactive or RSMRST# asserts	No	No	Yes (Note 2)
Special shutdown cycle from CPU causes CF9h-like PLTRST# and CF9h Global Reset Bit = 1	No	No	Yes
Special shutdown cycle from CPU causes CF9h-like PLTRST# and CF9h Bit 3 = 1	No	Yes	No (Note 2)
Special shutdown cycle from CPU causes CF9h-like PLTRST# and CF9h Global Reset Bit = 0	Yes	No	No (Note 1)
Intel® Management Engine Triggered Host Reset without power cycle	Yes	No	No (Note 1)
Intel® Management Engine Triggered Host Reset with power cycle	No	Yes	No (Note 1)
Intel® Management Engine Triggered Global Reset	No	No	Yes
Intel® Management Engine Initiated Host Reset with power down	No	Yes (Note 3)	No (Note 1)
PLTRST# Entry Timeout	No	No	Yes
Intel® Management Engine Watchdog Timer	No	Yes (Note 4)	No (Note 1)
Power Management Watchdog Timer	No	Yes (Note 4)	No (Note 1)
CPUPWRGD Stuck Low	No	No	Yes
CPUPWRGD-to-CPURST# Violation	No	No	Yes

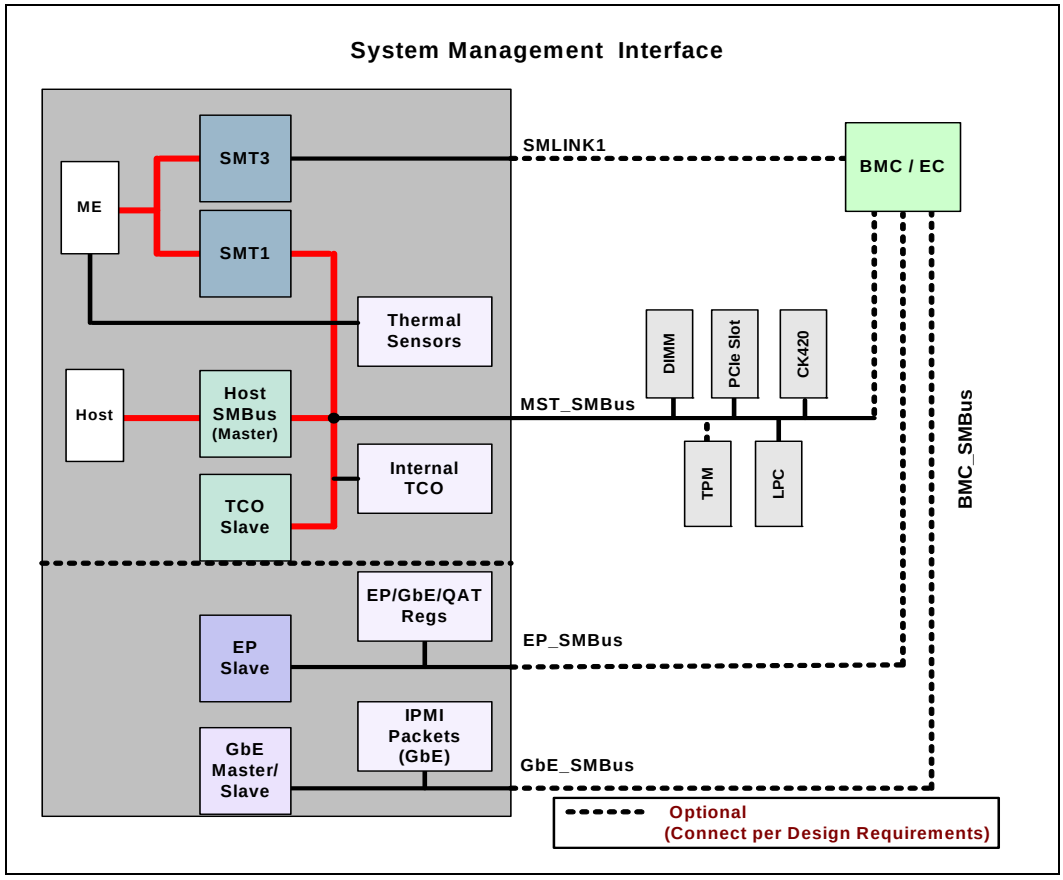
Notes:
1. Trigger results in Global Reset with power cycle if the acknowledge message is not received.
2. The system does not send warning message to CPU, reset occurs without delay.
3. The system waits for enabled wake event to complete reset.
4. The system allowed to drop this type of reset request if received while the system is in S3/S4/S5.

The system must not drop this type of reset request if received while system is in a software-entered S3/S4/S5 state. However, the system is allowed to perform the reset without executing the RESET_WARN protocol in these states.

4.12 System Management

The System Management Bus (SMBus) 2.0 Host Controller and SMBus Slave Interfaces make it easier to manage and lower Total Cost of Ownership (TCO) of the system. Figure 4-4 provides a high-level block of the System Management Interfaces. The following sections provides a description of the functions and protocols of the SMBus interfaces.

Figure 4-4. System Management Bus (SMBus) Interface



4.12.1 SMBus Host Controller (B0:D31:F3)

The host controller provides a mechanism for the processor to initiate communications with SMBus peripherals (slaves). The Host Controller is also capable of operating in a mode in which it can communicate with I^2C compatible devices.

See Section 11.0, "SMBus Controller Registers (B0:D31:F3)" for SMBus Host Controller Configuration register descriptions.

SMBus messages can be performed with either packet error checking (PEC) enabled or disabled. The actual PEC calculation and checking is performed in hardware.

The Slave Interface allows an external master to take control of the bus and perform read or write transactions. Write cycles can be used to cause certain events or pass messages, and the read cycles can be used to determine the state of various status bits. The system internal host controller cannot access the internal Slave Interface.

The SMBus exists in Bus 0:Device 31:Function 3 configuration space, and consists of a transmit data path, and host controller. The transmit data path provides the data flow logic needed to implement the seven different SMBus command protocols and is controlled by the host controller. The SMBus host controller logic is clocked by RTC clock.

The SMBus Address Resolution Protocol (ARP) is supported by using the existing host controller commands through software, except for the new Host Notify command (which is actually a received message).

The programming model of the host controller is combined into two portions: a PCI configuration portion, and a system I/O mapped portion. All static configuration, such as the I/O base address, is done via the PCI configuration space. Real-time programming of the Host interface is done in system I/O space.

The SMBus host controller checks for parity errors as a target. If an error is detected, the detected parity error bit in the PCI Status Register (Device 31:Function 3:Offset 06h:bit 15) is set. If bit 6 and bit 8 of the PCI Command Register (Device 31:Function 3:Offset 04h) are set, an SERR# is generated and the signaled SERR# bit in the PCI Status Register (bit 14) is set.

4.12.1.1 Host Controller Interface

The SMBus host controller is used to send commands to other SMBus slave devices. Software sets up the host controller with an address, command, and, for writes, data and optional PEC; and then tells the controller to start. When the controller has finished transmitting data on writes, or receiving data on reads, it generates an SMI# or interrupt, if enabled.

The host controller supports 8 command protocols of the SMBus interface (See *System Management Bus (SMBus) Specification, Version 2.0*): Quick Command, Send Byte, Receive Byte, Write Byte/Word, Read Byte/Word, Process Call, Block Read/Write, Block Write–Block Read Process Call, and Host Notify.

The SMBus host controller requires that the various data and command fields be setup for the type of command to be sent. When software sets the START bit, the SMBus Host controller performs the requested transaction, and interrupts the processor (or generates an SMI#) when the transaction is completed. Once a START command has been issued, the values of the “active registers” (Host Control, Host Command, Transmit Slave Address, Data 0, Data 1) should not be changed or read until the interrupt status message (INTR) has been set (indicating the completion of the command). Any register values needed for computation purposes should be saved prior to issuing of a new command, as the SMBus host controller updates all registers while completing the new command.

The Host Controller supports the *System Management Bus (SMBus) Specification, Version 2.0*. Slave functionality, including the Host Notify protocol, is available on the SMBus pins.

Note: Using the SMBus host controller to send commands to any SMBus slave port is not supported.

4.12.1.2 Command Protocols

In all of the following commands, the Host Status Register (offset 00h) is used to determine the progress of the command. While the command is in operation, the HOST_BUSY bit is set. If the command completes successfully, the INTR bit is set in the Host Status Register. If the device does not respond with an acknowledge, and the transaction times out, the DEV_ERR bit is set. If software sets the KILL bit in the Host Control Register while the command is running, the transaction will stop and the FAILED bit is set.

Quick Command

When programmed for a Quick Command, the Transmit Slave Address Register is sent. The PEC byte is never appended to the Quick Protocol. Software should force the PEC_EN bit to 0 when performing the Quick Command. Software must force the I2C_EN bit to 0 when running this command. See section 5.5.1 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Send Byte / Receive Byte

For the Send Byte command, the Transmit Slave Address and Device Command Registers are sent. For the Receive Byte command, the Transmit Slave Address Register is sent. The data received is stored in the DATA0 register. Software must force the I2C_EN bit to 0 when running this command.

The Receive Byte is similar to a Send Byte, the only difference is the direction of data transfer. See sections 5.5.2 and 5.5.3 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Write Byte/Word

The first byte of a Write Byte/Word access is the command code. The next 1 or 2 bytes are the data to be written. When programmed for a Write Byte/Word command, the Transmit Slave Address, Device Command, and Data0 Registers are sent. In addition, the Data1 Register is sent on a Write Word command. Software must force the I2C_EN bit to 0 when running this command. See section 5.5.4 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Read Byte/Word

Reading data is slightly more complicated than writing data. First, the host must write a command to the slave device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The slave then returns 1 or 2 bytes of data. Software must force the I2C_EN bit to 0 when running this command.

When programmed for the read byte/word command, the Transmit Slave Address and Device Command Registers are sent. Data is received into the DATA0 on the read byte, and the DAT0 and DATA1 registers on the read word. See section 5.5.5 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Process Call

The process call is so named because a command sends data and waits for the slave to return a value dependent on that data. The protocol is simply a Write Word followed by a Read Word, but without a second command or stop condition.

When programmed for the Process Call command, the host transmits the Transmit Slave Address, Host Command, DATA0 and DATA1 registers. Data received from the device is stored in the DATA0 and DATA1 registers. The Process Call command with

I2C_EN set and the PEC_EN bit set produces undefined results. Software must force either I2C_EN or PEC_EN to 0 when running this command. See section 5.5.6 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Note: For process call command, the value written into bit 0 of the Transmit Slave Address Register (SMB I/O register, offset 04h) needs to be 0.

- If the I2C_EN bit is set, the protocol sequence changes slightly: the Command Code (bits 18:11 in the bit sequence) are not sent. As a result, the slave does not acknowledge (bit 19 in the sequence).

Block Read/Write

The system contains a 32-byte buffer for read and write data which can be enabled by setting bit 1 of the Auxiliary Control register at offset 0Dh in I/O space, as opposed to a single byte of buffering. This 32-byte buffer is filled with write data before transmission, and filled with read data on reception. The interrupt is generated only after a transmission or reception of 32 bytes, or when the entire byte count has been transmitted/received.

The byte count field is transmitted but ignored as software will end the transfer after all bytes it cares about have been sent or received.

For a Block Write, software must either force the I2C_EN bit or both the PEC_EN and AAC bits to 0 when running this command.

The block write begins with a slave address and a write condition. After the command code, a byte count is issued to indicate how many more bytes will follow in the message. If a slave had 20 bytes to send, the first byte would be the number 20 (14h), followed by 20 bytes of data. The byte count may not be 0. A Block Read or Write is allowed to transfer a maximum of 32 data bytes.

When programmed for a block write command, the Transmit Slave Address, Device Command, and Data0 (count) registers are sent. Data is then sent from the Block Data Byte register; the total data sent being the value stored in the Data0 Register. On block read commands, the first byte received is stored in the Data0 register, and the remaining bytes are stored in the Block Data Byte register. See section 5.5.7 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

Note: For Block Write, if the I2C_EN bit is set, the format of the command changes slightly. The host will still send the number of bytes (on writes) or receive the number of bytes (on reads) indicated in the DATA0 register. However, it does not send the contents of the DATA0 register as part of the message. Also, the Block Write protocol sequence changes slightly: the Byte Count (bits 27:20 in the bit sequence) are not sent - as a result, the slave does not send an acknowledge (bit 28 in the sequence).

I^2C Read

This command allows the interface to perform block reads to certain I^2C devices, such as serial E^2PROMs. The SMBus Block Read supports the 7-bit addressing mode only.

However, this does not allow access to devices using the I^2C “Combined Format” that has data bytes after the address. Typically these data bytes correspond to an offset (address) within the serial memory chips.

Note: This command is supported independent of the setting of the I2C_EN bit. The I^2C Read command with the PEC_EN bit set produces undefined results. Software must force both the PEC_EN and AAC bit to 0 when running this command.

For I^2C Read command, the value written into bit 0 of the Transmit Slave Address Register (SMB I/O register, offset 04h) needs to be 0.

The format that is used for the command is shown in Table 4-33.

Table 4-33. I^2C Block Read

Bit	Description
1	Start
8:2	Slave Address — 7 bits
9	Write
10	Acknowledge from slave
18:11	Send DATA1 register
19	Acknowledge from slave
20	Repeated Start
27:21	Slave Address — 7 bits
28	Read
29	Acknowledge from slave
37:30	Data byte 1 from slave — 8 bits
38	Acknowledge
46:39	Data byte 2 from slave — 8 bits
47	Acknowledge
–	Data bytes from slave / Acknowledge
–	Data byte N from slave — 8 bits
–	NOT Acknowledge
–	Stop

The host will continue reading data from the peripheral until the NAK is received.

Block Write–Block Read Process Call

The block write-block read process call is a two-part message. The call begins with a slave address and a write condition. After the command code the host issues a write byte count (M) that describes how many more bytes is written in the first part of the message. If a master has 6 bytes to send, the byte count field has the value 6 (0000 0110b), followed by the 6 bytes of data. The write byte count (M) cannot be 0.

The second part of the message is a block of read data beginning with a repeated start condition followed by the slave address and a Read bit. The next byte is the read byte count (N), which may differ from the write byte count (M). The read byte count (N) cannot be 0.

The combined data payload must not exceed 32 bytes. The byte length restrictions of this process call are summarized as follows:

- $M \geq 1$ byte
- $N \geq 1$ byte
- $M + N \leq 32$ bytes

The read byte count does not include the PEC byte. The PEC is computed on the total message beginning with the first slave address and using the normal PEC computational rules. It is highly recommended that a PEC byte be used with the Block Write-Block Read Process Call. Software must do a read to the command register (offset 2h) to reset the 32 byte buffer pointer prior to reading the block data register.

There is no STOP condition before the repeated START condition, and a NACK signifies the end of the read transfer.

Note: E32B bit in the Auxiliary Control register must be set when using this protocol.

See section 5.5.8 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the protocol format.

4.12.1.3 Bus Arbitration

Several masters may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The host continuously monitors the SMBDATA line. When the host is attempting to drive the bus to a 1 by letting go of the SMBDATA line and it samples SMBDATA low, then some other master is driving the bus and the host stops transferring data.

If the host sees that it has lost arbitration, the condition is called a collision. The host sets the BUS_ERR bit in the Host Status Register, and if enabled, generate an interrupt or SMI#. The processor is responsible for restarting the transaction.

When the host is the SMBus master, it drives the clock. When the host is sending address or command as an SMBus master, or data bytes as a master on writes, it drives data relative to the clock it is also driving. It does not start toggling the clock until the start or stop condition meets proper setup and hold time. the host also guarantees minimum time between SMBus transactions as a master.

The system supports the same arbitration protocol for both the SMBus and the System Management Link (SMLink) interfaces.

4.12.1.4 Bus Timing

4.12.1.5 Clock Stretching

Some devices may not be able to handle their clock toggling at the rate that the host, as an SMBus master, would like. They have the capability of stretching the low time of the clock. When the host attempts to release the clock, allowing the clock to go high, the clock remains low for an extended period of time.

The host monitors the SMBus clock line after it releases the bus to determine whether to enable the counter for the high time of the clock. While the bus is still low, the high time counter must not be enabled. Similarly, the low period of the clock can be stretched by an SMBus master if it is not ready to send or receive data.

4.12.1.6 Bus Time Out (Host as SMBus Master)

If there is an error in the transaction such that an SMBus device does not signal an acknowledge or holds the clock lower than the allowed time-out time, the transaction times out. the host discards the cycle and set the DEV_ERR bit. The time out minimum is 25 ms (800 RTC clocks). The time-out counter inside the host starts after the last bit of data is transferred by the host and it is waiting for a response.

The 25 ms timeout counter does not count under the following conditions:

1. BYTE_DONE_STATUS bit (SMBus I/O Offset 00h, bit 7) is set
2. The SECOND_TO_STS bit (TCO I/O Offset 06h, bit 1) is not set (this indicates that the system has not locked up).

4.12.1.7 Interrupts / SMI#

The Host's SMBus controller uses PIRQB# as its interrupt pin. However, the system can alternatively be set up to generate SMI# instead of an interrupt, by setting the SMBUS_SMI_EN bit (Device 31:Function 0:Offset 40h:bit 1).

Table 4-34 and Table 4-35 specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and Slave SMI, and Wake internal signals. The rows in the tables are additive, which means that if more than one row is true for a particular scenario then the results for all of the activated rows occurs.

Table 4-34. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	SMBALERT_DIS (Slave Command I/O Register, Offset 11h, Bit 2)	Result
SMBALERT# asserted low (always reported in Host Status Register, Bit 5)	X	X	X	Wake generated
	X	1	0	Slave SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 4-35. Enables for SMBus Slave Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit1)	Event
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS).
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

Table 4-36. Enables for the Host Notify Command

HOST_NOTIFY_INTREN (Slave Control I/O Register, Offset 11h, bit 0)	SMB_SMI_EN (Host Config Register, D31:F3:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Slave Control I/O Register, Offset 11h, bit 1)	Result
0	X	0	None
X	X	1	Wake generated
1	0	X	Interrupt generated
1	1	X	Slave SMI# generated (SMBUS_SMI_STS)

4.12.1.8 SMBALERT#

SMBALERT# is multiplexed with GPIO[11]. When enable and the signal is asserted, the system can generate an interrupt, an SMI#, or a wake event from S1–S5.

4.12.1.9 SMBus CRC Generation and Checking

If the AAC bit is set in the Auxiliary Control register, the host automatically calculates and drives CRC at the end of the transmitted packet for write cycles, and checks the CRC for read cycles. It does not transmit the contents of the PEC register for CRC. The PEC bit must not be set in the Host Control register if this bit is set or it results in unspecified behavior.

If the read cycle results in a CRC error, the DEV_ERR bit and the CRCE bit in the Auxiliary Status register at offset 0Ch is set.

4.12.2 TCO Slave SMBus Interface

The internal TCO Slave shares the external pins with the Host SMBus. The TCO Slave SMBus logic does not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol device. The slave interface decodes address cycles, and allows an external microcontroller to perform specific actions. The TCO Slave is connected internally to the host SMBus by setting the soft trap TCO Slave Select in the flash descriptor.

Key features and capabilities of the TCO Slave include:

- Supports decode of three types of messages: Byte Write, Byte Read, and Host Notify.
- Receive Slave Address register: This is the address that the slave decodes. A default value is provided so that the slave interface can be used without the processor having to program this register.
- Receive Slave Data register in the SMBus I/O space that includes the data written by the external microcontroller.
- Registers that the external microcontroller can read to get the state of the system.
- Status bits to indicate that the SMBus slave logic caused an interrupt or SMI# due to the reception of a message that matched the slave address:
  - Bit 0 of the Slave Status Register for the Host Notify command
  - Bit 16 of the SMI Status Register for all others

The external microcontroller should not attempt to access the TCO SMBus slave logic until either:

- One second after both: RTCRST# is high and RSMRST# is high, OR
- PLTRST# de-asserts

If a master leaves the clock and data bits of the SMBus interface at 1 for 50 µs or more in the middle of a cycle, the slave logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the slave logic.

Note: When an external microcontroller accesses the SMBus Slave Interface over the SMBus a translation in the address is needed to accommodate the least significant bit used for read/write control. For example, if the slave address (RCV_SLVA) is left at 44h (default), the external micro controller would use an address of 88h/89h (write/read).

4.12.2.1 Format of Slave Write Cycle

The external master performs Byte Write commands to the SMBus Slave I/F. The "Command" field (bits 11:18) indicate which register is being accessed. The Data field (bits 20:27) indicate the value that should be written to that register.

Table 4-37 provides the values associated with the Slave Write registers.

Table 4-37. Slave Write Registers

Register	Function
0	Command Register. See Table 4-38, "Command Types" for legal values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–7	Reserved
8	Reserved
9–FFh	Reserved

Note: The external microcontroller is responsible for making sure it does not update the contents of the data byte registers until they have been read by the system processor. The system overwrites the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. The system does not attempt to cover this race condition (for example, unpredictable results in this case).

Table 4-38. Command Types

Command Type	Description
0	Reserved
1	WAKE/SMI# - This command wakes the system if it is not already awake. If system is already awake, an SMI# is generated. ***Note:*** The SMB_WAK_STS bit is set by this command, even if the system is already awake. The SMI handler should then clear this bit.
2	Unconditional Powerdown - This command sets the PWRBTNOR_STS bit, and has the same effect as the Power button Override occurring.
3	HARD RESET WITHOUT CYCLING - This command causes a hard reset of the system (does not include cycling of the power supply). This is equivalent to a write to the CF9h register with bits 2:1 set to 1, but bit 3 set to 0.
4	HARD RESET SYSTEM - This command causes a hard reset of the system (including cycling of the power supply). This is equivalent to a write to the CF9h register with bits 3:1 set to 1.
5	Disable the TCO Messages - This command will disable the Slave from sending Heartbeat and Event messages (as described in Section 4.12). Once this command has been executed, Heartbeat and Event message reporting can only be re-enabled by assertion and deassertion of the RSMRST# signal.
6	WD RELOAD - Reload watchdog timer.
7	Reserved
8	SMLINK_SLV_SMI - When the Slave detects this command type while in the S0 state, it sets the SMLINK_SLV_SMI_STS bit. This command should only be used if the system is in an S0 state. If the message is received during S1–S5 states, the system acknowledges it, but the SMLINK_SLV_SMI_STS bit does not get set. ***Note:*** It is possible that the system transitions out of the S0 state at the same time that the SMLINK_SLV_SMI command is received. In this case, the SMLINK_SLV_SMI_STS bit may get set but not serviced before the system goes to sleep. Once the system returns to S0, the SMI associated with this bit would then be generated. Software must be able to handle this scenario.
9-FFh	Reserved.

4.12.2.2 Format of Read Command

The external master performs Byte Read commands to the TCO SMBus Slave interface. The Command field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 4-39. Slave Read Cycle Format

Bit	Description	Driven by	Comment
1	Start	External Microcontroller	
2-8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
9	Write	External Microcontroller	Always 0
10	ACK	Slave	
11-18	Command code – 8 bits	External Microcontroller	Indicates which register is being accessed. See Table 4-38 for list of implemented registers.
19	ACK	Slave	
20	Repeated Start	External Microcontroller	
21-27	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
28	Read	External Microcontroller	Always 1
29	ACK	Slave	

Table 4-39. Slave Read Cycle Format

Bit	Description	Driven by	Comment
30-37	Data Byte	Slave	Value depends on register being accessed. See Table 4-40 for list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 4-40. Data Values for Slave Read Registers (Sheet 1 of 2)

Register	Bits	Description
0	07:00	Reserved for capabilities indication. Should always return 00h. Future chips may return another value to indicate different capabilities.
1	02:00	System Power State: 000 = S0 001 = S1 010 = Reserved 011 = S3 100 = S4 101 = S5 110 = Reserved 111 = Reserved
	07:03	Reserved
2	03:00	Reserved
	07:04	Reserved
3	05:00	Watchdog Timer current value. Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the system will always report 3Fh in this field.
	07:06	Reserved
4	00	1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	01	1 = BTI Temperature Event occurred. This bit is set if the system's THRM# input signal is active. Else this beat will read "0."
	02	DOA CPU Status. This bit is 1 to indicate that the processor is dead
	03	1 = SECOND_TO_STS bit set. This bit is set after the second time-out (SECOND_TO_STS bit) of the Watchdog Timer occurs.
	06:04	Reserved. Will always be 0, but software should ignore.
	07	Reflects the value of the GPIO[11]/SMBALERT# pin (and is dependent upon the value of the GPI_INV[11] bit. If the GPI_INV[11] bit is 1, then the value in this bit equals the level of the GPI[11]/SMBALERT# pin (high = 1, low = 0). If the GPI_INV[11] bit is 0, then the value of this bit will equal the inverse of the level of the GPIO[11]/SMBALERT# pin (high = 0, low = 1).

Table 4-40. Data Values for Slave Read Registers (Sheet 2 of 2)

Register	Bits	Description
5	00	Reserved
	01	Reserved
	02	CPU Power Failure Status: - '1' if the CPUPWR_FLR bit in the GEN_PMCON_2 register is set.
	03	INIT3_3V# due to receiving Shutdown message - This event is visible from the reception of the shutdown message until a platform reset is done if the Shutdown Policy Select bit (SPS) is configured to drive INIT3_3V#. When the SPS bit is configured to generate PLTRST# based on shutdown, this register bit will always return 0. Events on signal do not create a event message
	04	Reserved
	05	POWER_OK_BAD - Indicates the failure core power well ramp during boot/resume. This bit is active if the SLP_S3# pin is de-asserted and PWROK pin is not asserted.
	06	Thermal Trip - This bit will shadow the state of processor Thermal Trip status bit (CTS) (16.2.1.2, GEN_PMCON_2, bit 3). Events on signal does not create an event message.
	07	Reserved - Default value is "X" Note: Software should not expect a consistent value when this bit is read through SMBUS/SMLink
6	07:00	Contents of the Message 1 register. See Section 6.8.1.8 for the description of this register.
7	07:00	Contents of the Message 2 register. See Section 6.8.1.8 for the description of this register.
8	07:00	Contents of the TCO_WDCNT register. See Section 6.8.1.9 for the description of this register.
9	07:00	Seconds of the RTC
A	07:00	Minutes of the RTC
B	07:00	Hours of the RTC
C	07:00	"Day of Week" of the RTC
D	07:00	"Day of Month" of the RTC
E	07:00	Month of the RTC
F	07:00	Year of the RTC
10h–FFh	07:00	Reserved

4.12.2.3 Behavioral Notes

According to the SMBus protocol, Read and Write messages begin with a Start bit – Address– Write bit sequence. When the Slave detects that the address matches the value in the Receive Slave Address register, it assumes that the protocol is always followed and ignores the Write bit (bit 9) and signal an Acknowledge during bit 10. In other words, if a Start–Address–Read occurs (which is illegal for SMBus Read or Write protocol), and the address matches TCO Slave Address, the Slave still grabs the cycle.

Also according to SMBus protocol, a Read cycle contains a Repeated Start–Address–Read sequence beginning at bit 20. If the Address matches the Slave's Receive Slave Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Slave Read cycle.

Note: An external microcontroller must not attempt to access SMBus TCO Slave logic until at least 1 second after both RTCRST# and RSMRST# are deasserted (high).

4.12.2.4 Slave Read of RTC Time Bytes

The TCO SMBus slave interface allows external SMBus master to read the internal RTC's time byte registers.

The RTC time bytes are internally latched by the system's hardware whenever RTC time is not changing and SMBus is idle. This ensures that the time byte delivered to the slave read is always valid and it does not change when the read is still in progress on the bus. The RTC time will change whenever hardware update is in progress, or there is a software write to the RTC time bytes.

The SMBus slave interface only supports Byte Read operation. The external SMBus master will read the RTC time bytes one after another. Software must check and manage the possible time rollover when subsequent time bytes are read.

For example, assuming the RTC time is 11 hours, 59 minutes, 59 seconds: When the external SMBus master reads the hour as 11, and then reads the minute, it is possible that the rollover happens between the reads and the minute is read as 0. This results in 11 hours, 0 minutes instead of the correct time of 12 hours, 0 minutes. Unless it is certain that rollover does not occur, software is required to detect the possible time rollover by reading multiple times such that the read time bytes can be adjusted accordingly if needed.

4.12.2.5 Format of Host Notify Command

The system tracks and responds to the standard Host Notify command as specified in the *System Management Bus (SMBus) Specification,* Version 2.0. The host address for this command is fixed to 0001000b. If the system already has data for a previously-received host notify command which has not been serviced yet by the host software (as indicated by the HOST_NOTIFY_STS bit), then it NACKs following the host address byte of the protocol. This allows the host to communicate non-acceptance to the master and retain the host notify address and data values for the previous cycle until host software completely services the interrupt.

Note: Host software must always clear the HOST_NOTIFY_STS bit after completing any necessary reads of the address and data registers.

Table 4-41. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Master	
8:2	SMB Host Address — 7 bits	External Master	Always 0001_000
9	Write	External Master	Always 0
10	ACK (or NACK)	Slave	Slave NACKs if HOST_NOTIFY_STS is 1
17:11	Device Address – 7 bits	External Master	Indicates the address of the master; loaded into the Notify Device Address Register
18	Unused — Always 0	External Master	7-bit-only address; this bit is inserted to complete the byte
19	ACK	Slave	
27:20	Data Byte Low — 8 bits	External Master	Loaded into the Notify Data Low Byte Register
28	ACK	Slave	
36:29	Data Byte High — 8 bits	External Master	Loaded into the Notify Data High Byte Register
37	ACK	Slave	
38	Stop	External Master	

4.12.2.6 TCO Slave Functions

The TCO Slave can be assessed by an external microcontroller.

The following features and functions are supported by the TCO Slave:

- Processor present detection: Detects if processor fails to fetch the first instruction after reset
- Various Error detection (such as ECC Errors) indicated by host controller: Can generate SMI#, SCI, SERR, NMI, or TCO interrupt
- Intruder Detect input:
  - Can generate TCO interrupt or SMI# when the system cover is removed
  - INTRUDER# allowed to go active in any power state, including G3
- Detection of bad BIOS Flash (Flash on SPI) programming: Detects if data on first read is FFh (indicates that BIOS flash is not programmed)
- Ability to hide a PCI device: Allows software to hide a PCI device in terms of configuration space through the use of a device hide register.

4.12.2.7 Theory of Operation

The System Management functions are designed to allow the system to diagnose failing subsystems. The intent of this logic is that some of the system management functionality can be provided without the aid of an external microcontroller.

4.12.2.7.1 Detecting a System Lockup

When the processor is reset, it is expected to fetch its first instruction. If the processor fails to fetch the first instruction after reset, the TCO timer times out twice and the system asserts PLTRST#.

4.12.2.7.2 Handling an Intruder

The system has an input signal, INTRUDER#, that can be attached to a switch that is activated by the system's case being open. This input has a two RTC clock debounce. If INTRUDER# goes active (after the debouncer), this will set the INTRD_DET bit in the TCO_STS register. The INTRD_SEL bits in the TCO_CNT register can cause an SMI# or interrupt. The BIOS or interrupt handler can then cause a transition to the S5 state by writing to the SLP_EN bit.

The software can also directly read the status of the INTRUDER# signal (high or low) by clearing and then reading the INTRD_DET bit. This allows the signal to be used as a GPI if the intruder function is not required.

If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET signal will go to a 0 when INTRUDER# input signal goes inactive. This is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

Note: The INTRD_DET bit resides in the RTC well, and is set and cleared synchronously with the RTC clock. Thus, when software attempts to clear INTRD_DET (by writing a 1 to the bit location) there may be as much as two RTC clocks (about 65 µs) delay before the bit is actually cleared. The INTRUDER# signal should be asserted for a minimum of 1 ms to guarantee that the INTRD_DET bit is set.

- If the INTRUDER# signal is still active when software attempts to clear the INTRD_DET bit, the bit remains set and the SMI is generated again immediately. The SMI handler can clear the INTRD_SEL bits to avoid further SMIs. However, if

the INTRUDER# signal goes inactive and then active again, there is not further SMIs, since the INTRD_SEL bits would select that no SMI# be generated.

4.12.2.7.3 Detecting Improper Flash Programming

The system can detect the case where the BIOS flash is not programmed. This results in the first instruction fetched to have a value of FFh. If this occurs, the system sets the BAD_BIOS bit. The BIOS flash resides in SPI flash.

4.12.2.7.4 TCO Trigger Events

Table 4-42 provides the events that can trigger TCO messages. The external BMC has to poll to detect pending messages.

Table 4-42. Event Transitions that Cause Messages

Event	Assertion?	Deassertion?	Comments
INTRUDER# pin	yes	no	Must be in "S1 or hung S0" state
THRM# pin	yes	yes	Must be in "S1 or hung S0" state. The THRM# pin is isolated when the core power is off, thus preventing this event in S3-S5.
Watchdog Timer Expired	yes	no (NA)	"S1 or hung S0" state entered
GPIO[11]/SMBALERT# pin	yes	yes	Must be in "S1 or hung S0" state
BATLOW#	yes	yes	Must be in "S1 or hung S0" state
CPU_PWR_FLR	yes	no	"S1 or hung S0" state entered

4.12.3 EndPoint (EP) Slave SMBus

This SMBus slave port provides system management (SM) visibility into all configuration registers in the EndPoint.

The EP SMBus operations may be split into two upper level protocols: writing information to configuration registers and reading configuration registers. This section describes the required protocol for an SMBus master to access the EP's internal configuration registers. See the *SMBus Specification*, Revision 2.0 for the specific bus protocol, timings, and waveforms.

EP SMBus Features:

- The EP SMBus allows access to any register within the EP whether the CSR exists in PCI (bus, device, function) space or in memory mapped space.
- The EP SMBus interface acts as a side-band configuration access to the internal configuration space, therefore, the config block must service all SMBus config transactions even in the presence of a deadlock condition where the system cannot respond to configuration requests on the PCIe* links.
- The EP SMBus supports Packet Error Checking (can be disabled) as defined in the SMBus 2.0 specification.
- The EP SMBus requires the SMBus master to poll the busy bit to determine if the previous transaction has completed. For reads, this is after the repeated start sequence.

4.12.3.1 SMBus Supported Transactions

The EP SMBus supports six SMBus individual commands associated into two groups (read/write) with three data sizes. The read transactions require two SMBus sequences, one is writing the requested read address to the internal register stack and the other is the read command to extract the data once it is available. The write transactions are a single sequence that contains both the address and the data.

Supported transactions:

- Block Write (Dword sized data packet)
- Block Read (Dword sized data packet)
- Word Write (Word sized data packet)
- Word Read (Word sized data packet)
- Byte Write (Byte sized data packet)
- Byte Read (Byte sized data packet)

To support longer PCIe* timeouts the SMBus master is required to poll the busy bit to know when the data in the stack contains the desired data. This applies to both reads and writes. The protocol diagrams (Figure 4-6 through Figure 4-16) only shows the polling in read transactions. The reason for this is due to the length of PCIe* timeouts which may be as long as several seconds. This violates the SMBus spec of a maximum of 25 ms. To overcome this limitation, the SMBus slave requests the configuration master for access. Once granted, the slave asserts its busy bit and releases the link. The SMBus master is free to address other devices on the link or poll the busy bit until any pending transaction is completed.

The command format is illustrated in Table 4-43 and the subsequent bulleted list of sub-field encodings.

Table 4-43. SMBus Command Encoding

7	6	5	4	3:2	1:0
Begin	End	MemTrans	PEC_en	Internal Command: 00 - Read DWord 01 - Write Byte 10 - Write Word 11 - Write DWord	SMBus command: 00 - Byte 01 - Word 10 - Block 11 - Reserved. Block command is selected.

- The *Begin* bit indicates the first transaction of the read or write sequence.
- The *End* bit indicates the last transaction of the read or write sequence.
- The *MemTrans* bit indicates the configuration request is a memory mapped addressed register or a PCI (bus, device, function, offset) addressed register. A logic 0 addresses a PCI configuration register. A logic 1 addresses a memory mapped register. When this bit is set it enables the designation memory address type.
- The *PEC_en* bit enables the 8-bit packet error checking (PEC) generation and checking logic. For the examples below, if PEC was disabled, then there would be no PEC generated or checked by the slave.
- The *Internal Command* field specifies the internal command to be issued by the SMBus slave logic. The EP SMBus supports dword reads and byte, word, and dword writes to configuration space.

- The *SMBus Command* field specifies the SMBus command to be issued on the bus. This field is used as an indication of the length of transfer so that the slave knows when to expect the PEC packet (if enabled).

The SMBus interface uses an internal register stack that is filled by the SMBus master before a request to the config master block is made. Table 4-44 provides a list of the bytes in the stack and their descriptions.

Table 4-44. Internal SMBus Protocol Stack

SMBus Stack usage for bus/dev/func commands (cmd[5] = 0)	SMBus Stack usage for memory region commands (cmd[5] = 1)	Description
Command	Command	Command byte
Byte Count	Byte Count	The number of bytes for this transaction when Block command is used.
Bus Number	Memory region	Bus number for bus/dev config space command type. Memory region for memory config space command type.
Device/Function	Address [23:16]	Device[4:0] and Function[2:0] for cmd[5] = 0 type of config transaction. Address[23:16] for cmd[5] = 1 type of memory config transaction.
Address High	Address [15:8]	The following fields are further defined for cmd[5]=0: Address High[7:4] = Reserved[3:0] Address High [3:0] = Address[11:8]: This is the high order PCIe* address field. The following fields are further defined for cmd[5]=1: Address[15:8]
Register Offset	Address [7:0]	The following fields are further defined for cmd[5]=0: Lower order 8-bit register offset (Address[7:0]) The following fields are further defined for cmd[5]=1: Address [7:0]
Data3	Data3	Data byte 3
Data2	Data2	Data byte 2
Data1	Data1	Data byte 1
Data0	Data0	Data byte 0

4.12.3.2 Addressing

The EP SMBus slave address, which each component claims, is dependent on the GBE_LED0/EP_SMB_ADR2 and GBE_LED1/EP_SMB_ADR3 pin straps (sampled once VCCAEPAUX/VCCAEP_1P05 is stable. For the DH89xxCC this is on the assertion of GBE_AUX_PWR_OK). The EP SMBus slave is accessed with address [7:1] = 1110_XX0. The X's represent GBE_LED0/EP_SMB_ADR2 and GBE_LED1/EP_SMB_ADR3 strap pins. See Table 4-45 for the mapping of strap pins to the bit positions of the slave address.

Note: The slave address is dependent on the GBE_LED0/EP_SMB_ADR2 and GBE_LED1/EP_SMB_ADR3 strap pins only and cannot be reprogrammed.

Table 4-45. SMBus Slave Address Format

Slave Address Field Bit Position	Slave Address Source
[7]	1
[6]	1
[5]	1
[4]	0
[3]	GBE_LED1/EP_SMB_ADR3 strap pin
[2]	GBE_LED0/EP_SMB_ADR2 strap pin
[1]	0
[0]	Read/Write# bit. This bit is in the slave address field to indicate a read or write operation. It is not part of the SMBus slave address.

If the Mem/Cfg (MemTrans) bit as described in Table 4-43 is cleared then the address field represents the standard PCI register addressing nomenclature namely; bus, device, function and offset.

If the Mem/Cfg bit is set, the address field has a new meaning. Bits [23:0] hold a linear memory address and bits [31:24] is a byte to indicate which memory region it is. Table 4-46 describes the selections available. A logic one in a bit position enables that memory region to be accessed. If the destination memory byte is zero then no action is taken (no request is sent to the configuration master).

If a memory region address field is set to a reserved space, the RHSL slave performs the following actions:

- The transaction is not executed
- The slave releases the SCL (Serial Clock) signal.
- The master abort error status is set.

Table 4-46. Memory Region Address Field

Bit field	Memory Region Address Field
0Fh	Internal SMBus Register
0Eh-0Ah	Reserved
09h	GbE3/Reserved[1]
08h	GbE2/Reserved[a]
07h	GbE1/Reserved[a]
06h	GbE0/Reserved[a]
05h	Reserved
04h	Reserved
03h	Reserved
02h	Reserved
01h	EndPoint Memory-Mapped CSRs
00h	Reserved

1. SKU Dependent

4.12.3.3 SMBus Initiated Southbound Configuration Cycles

The platform SMBus master agent that is connected to the EP Slave SMBus can request a configuration transaction to a downstream PCI-Express* device.

4.12.3.4 SMBus Error Handling

SMBus Error Handling feature list:

- Errors are reported in the status byte field.
- Errors in Table 4-47 are also collected in the FERR and NERR registers.

The EP SMBus slave interface handles internal and PEC errors. For example, internal errors can occur when the system issues a configuration read on the PCI-Express* port and that read terminates in error. These errors manifest as a Not-Acknowledge (NACK) for the read command (*End* bit is set). If an internal error occurs during a configuration write, the final write command receives a NACK just before the stop bit. If the master receives a NACK, the entire configuration transaction should be reattempted.

If the master supports packet error checking (PEC) and the PEC_en bit in the command is set, then the PEC byte is checked in the slave interface. If the check indicates a failure, then the slave NACKs the PEC packet.

Each error bit must be routed to the FERR and NERR registers for error reporting. The status field encoding is defined in the Table 4-47. This field reports if an error occurred. If bits[2:0] are 000b then transaction was successful. A successful indication here does not necessarily mean that the transaction was completed correctly for all components in the system.

The busy bit is set whenever a transaction is accepted by the slave. This is true for reads and writes but the affects may not be observable for writes. This means that since the writes are posted and the communication link is so slow the master should never see a busy condition. A timeout is associated with the transaction in progress. When the timeout expires a timeout error status is asserted.

Table 4-47. Status Field Encoding for SMBus Reads

Bit	Description
07	Busy
06:03	Reserved
02:00	101-111: Reserved 100: Timeout Error 011: Master Abort. An error that is reported with respect to this transaction. 010: Completer Abort. An error is reported by downstream PCIe* device with respect to this transaction. 001: Memory Region encoding error. This bit is set if a memory region is not valid. 000: Successful

4.12.3.5 SMBus Interface Reset

The EP Slave interface state machine can be reset in several ways. The first two items are defined in the SMBus rev 2.0 specification.

- The master holds SCL low for 25 ms cumulative. Cumulative in this case means that all the "low time" for SCL is counted between the Start and Stop bit. If this totals 25 ms before reaching the Stop bit, the interface is reset.
- The master holds SCL continuously high for 50 us.
- Force a platform reset.

4.12.3.6 Configuration and Memory Read Protocol

Configuration and memory reads are accomplished through an SMBus write(s) and later followed by an SMBus read. The write sequence is used to initialize the Bus Number, Device, Function, and Register Number for the configuration access. The writing of this information can be accomplished through any combination of the supported SMBus write commands (Block, Word or Byte). The *Internal Command* field for each write should specify Read DWord.

After all the information is set up, the last write (*End* bit is set) initiates an internal configuration read. The slave asserts a busy bit in the status register and release the link with an acknowledge (ACK). The master SMBus performs the transaction sequence for reading the data, however, the master must observe the status bit [7] (busy) to determine if the data is valid. This is due to the PCIe* timeouts which may be long causing an SMBus spec violation. The SMBus master must poll the busy bit to determine when the pervious read transaction has completed.

If an error occurs then the status byte reports the results. This field indicates abnormal termination and contains status information such as target abort, master abort, and time-outs. Examples of configuration reads are illustrated below. All of these examples have PEC (Packet Error Code) enabled. If the master does not support PEC, then bit 4 of the command would be cleared and no PEC byte exists in the communication streams. For the definition of the diagram conventions below, see the *SMBus Specification*, Revision 2.0. For SMBus read transactions, the last byte of data (or the PEC byte if enabled) is NACKed by the master to indicate the end of the transaction.

4.12.3.6.1 SMBus Configuration and Memory Block-Size Reads

Figure 4-5. SMBus Block-Size Configuration Register Read

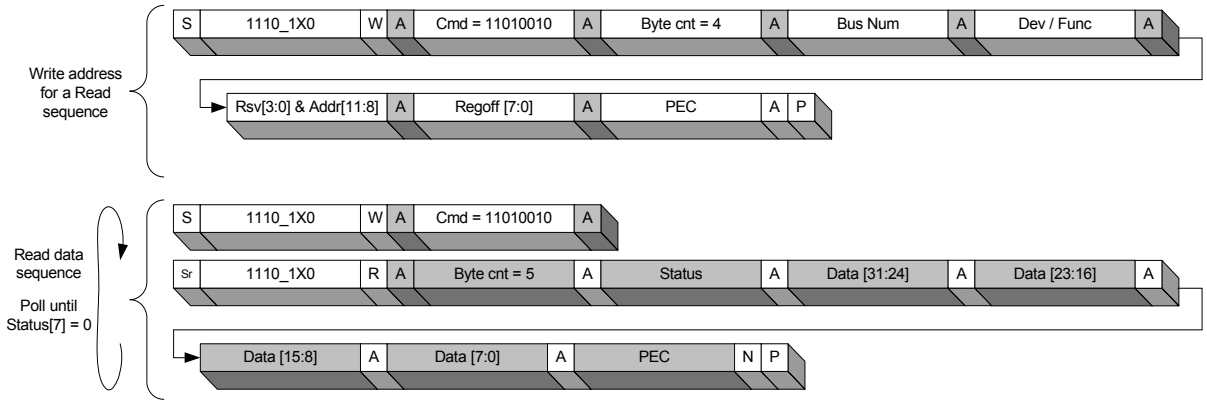


Figure 4-6. SMBus Block-Size Memory Register Read

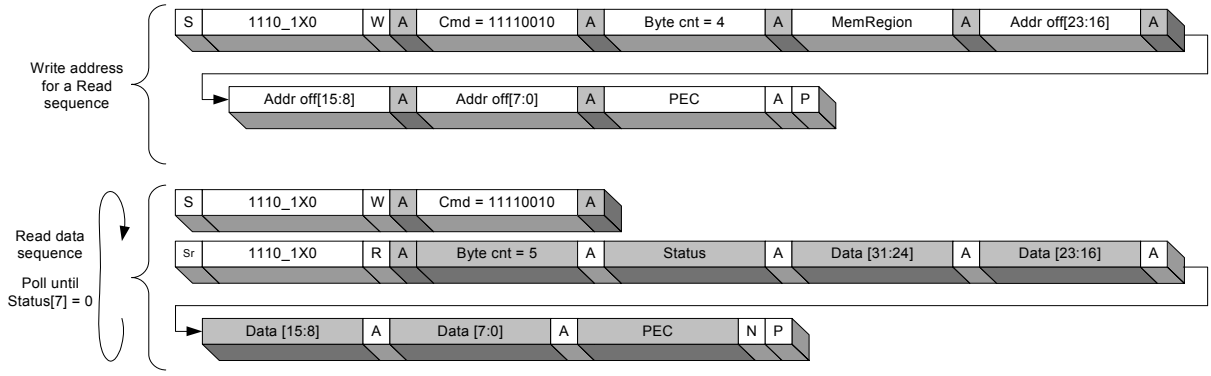


Figure 4-7. SMBus Word-Size Configuration Register Read

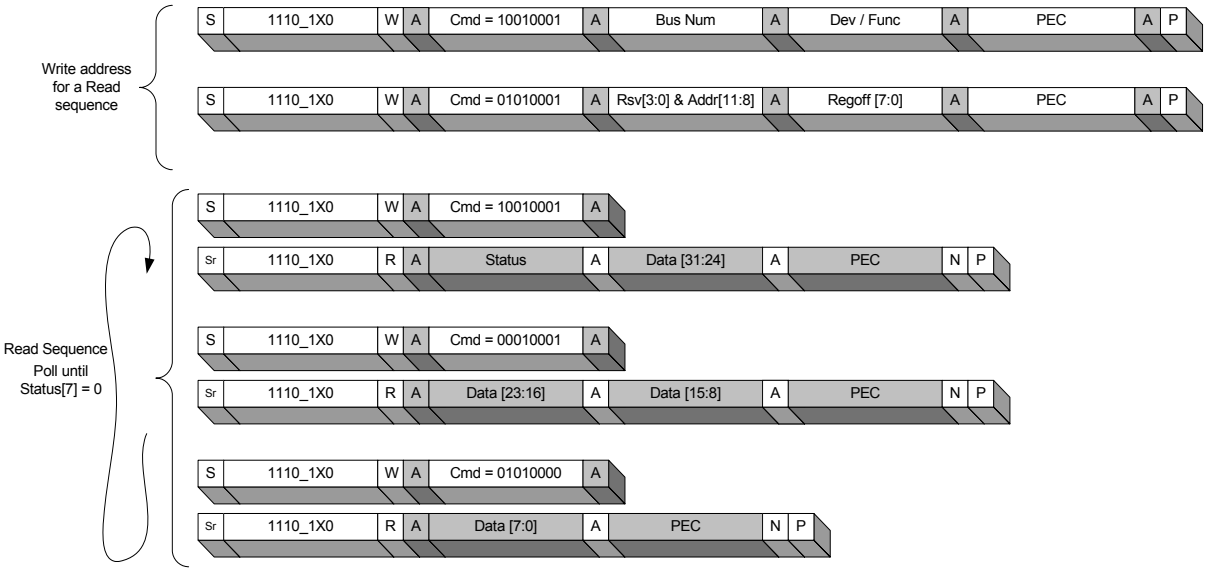
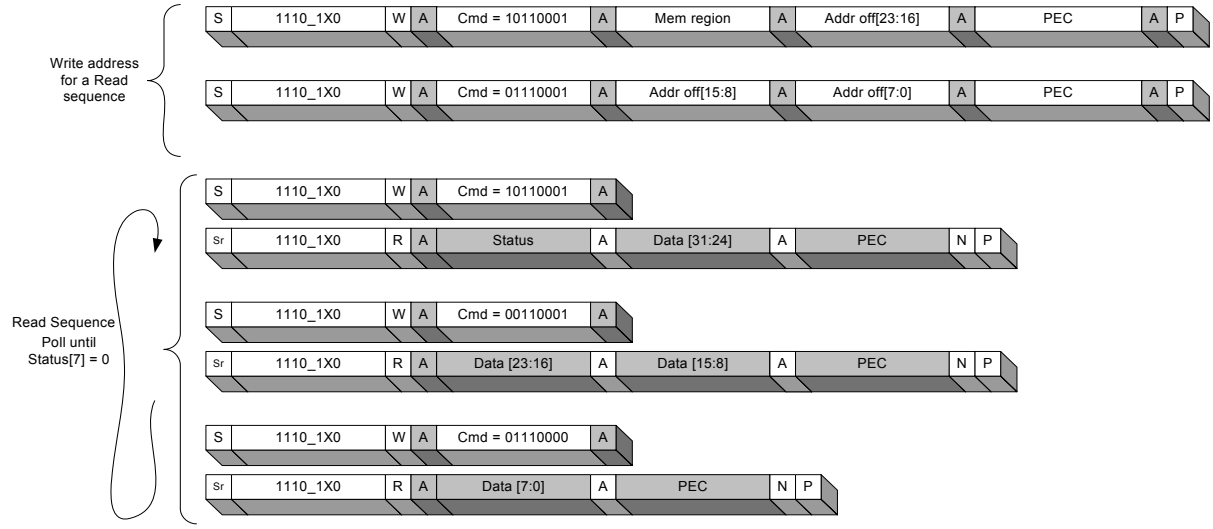


Figure 4-8. SMBus Word-Size Memory Register Read



4.12.3.6.2 SMBus Configuration and Memory Byte Reads

Figure 4-9. SMBus Byte-Size Configuration Register Read

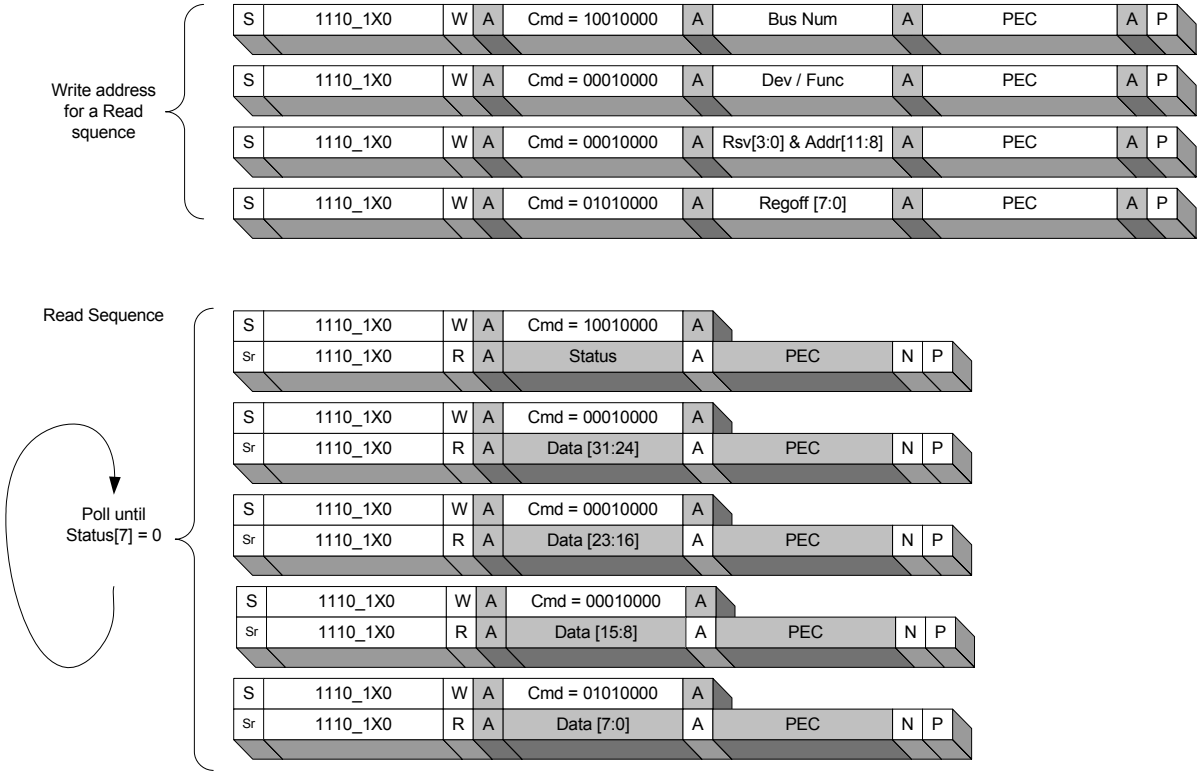
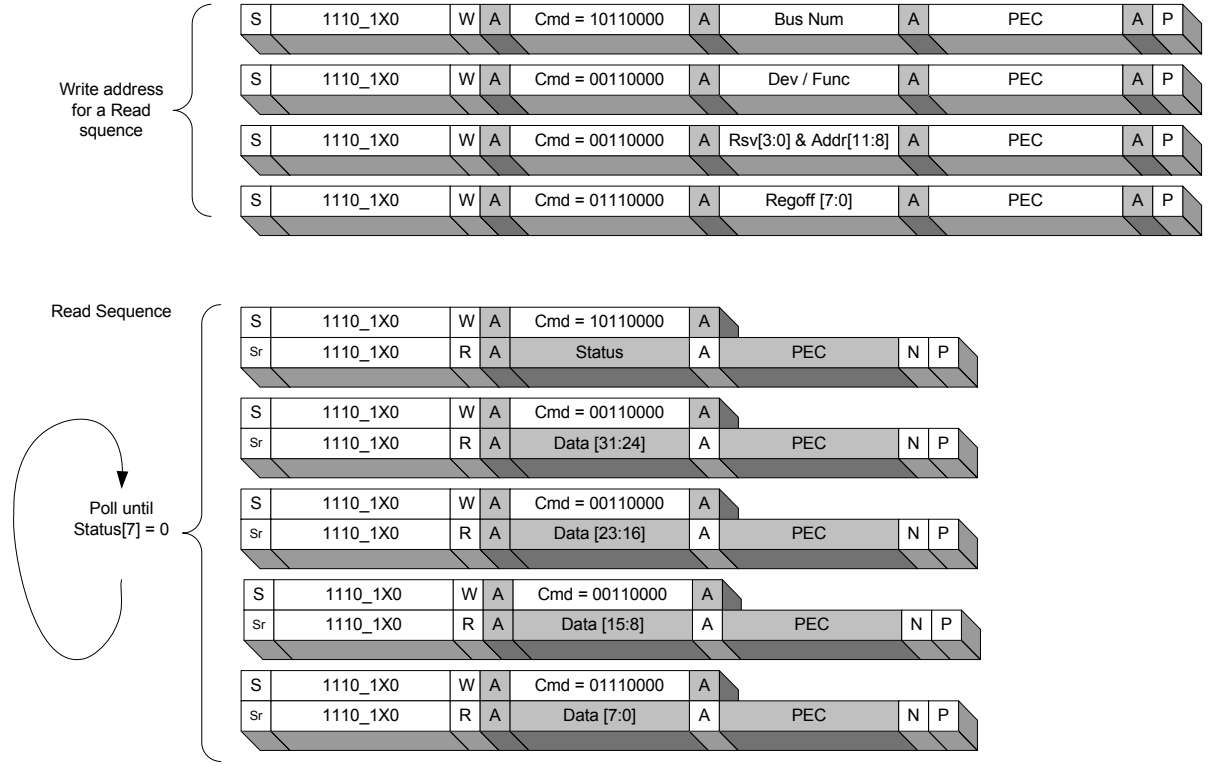


Figure 4-10. SMBus Byte-Size Memory Register Read



4.12.3.7 Configuration and Memory Write Protocol

Configuration and memory writes are accomplished through a series of SMBus writes. As with configuration reads, a write sequence is first used to initialize the Bus Number, Device, Function, and Register Number for the configuration access. The writing of this information can be accomplished through any combination of the supported SMBus write commands (Block, Word or Byte).

Note: The SMBus has no concept of byte enables. Therefore, the Register Number written to the slave is assumed to be aligned to the length of the Internal Command. In other words, for a Write Byte internal command, the Register Number specifies the byte address. For a Write DWord internal command, the two least-significant bits of the Register Number or Address Offset are ignored. This is different from PCI where the byte enables are used to indicate the byte of interest.

After all the information is set up, the SMBus master initiates one or more writes which sets up the data to be written. The final write (*End* bit is set) initiates an internal configuration write. The slave interface could potentially clock stretch the last data write until the write completes without error. If an error occurred, the SMBus interface NACKs the last write operation just before the stop bit.

The busy bit is set for the write transaction. A config write will likely complete before the SMBus master can poll the busy bit. If the transaction is destined to a chip on a PCIe* link then it could take several more clock cycle to complete the outbound transaction being sent.

Examples of configuration writes are illustrated below. For the definition of the diagram conventions below, see the *SMBus Specification*, Revision 2.0.

4.12.3.7.1 SMBus Configuration and Memory Block Writes

Figure 4-11. SMBus Block-Size Configuration Register Write

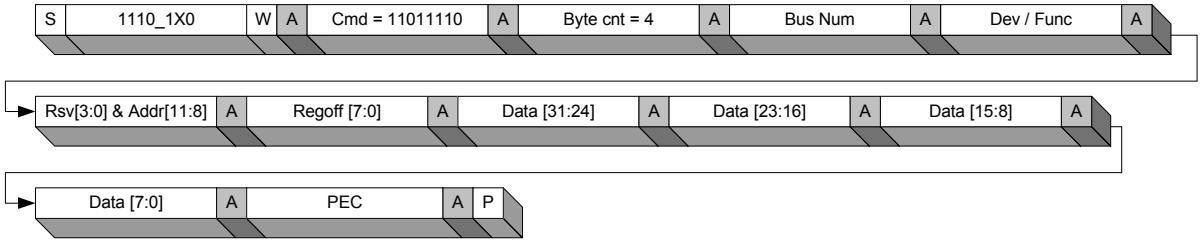
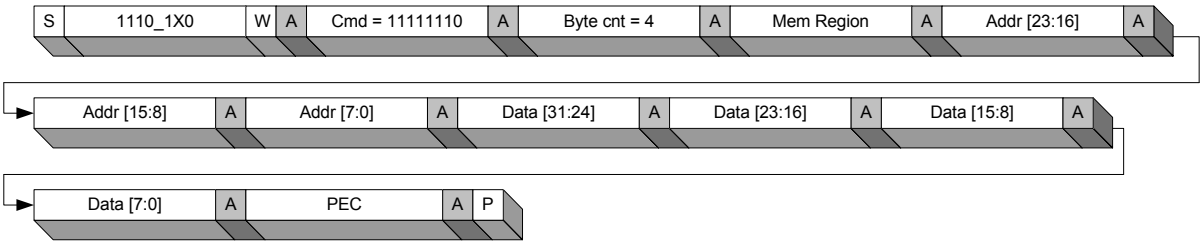


Figure 4-12. SMBus Block-Size Memory Register Write



4.12.3.7.2 SMBus Configuration and Memory Word Writes

Figure 4-13. SMBus Word-Size Configuration Register Write

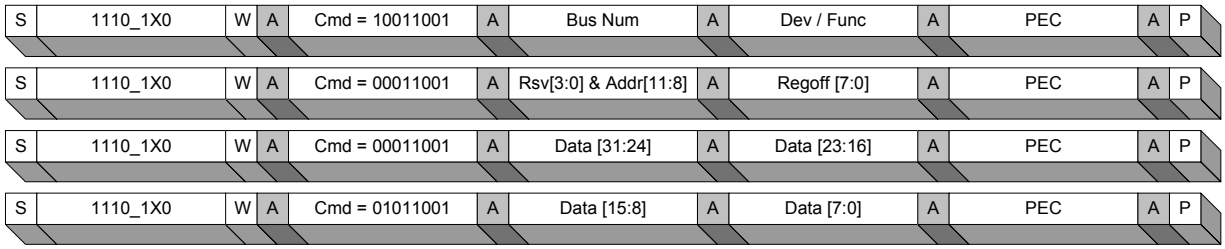
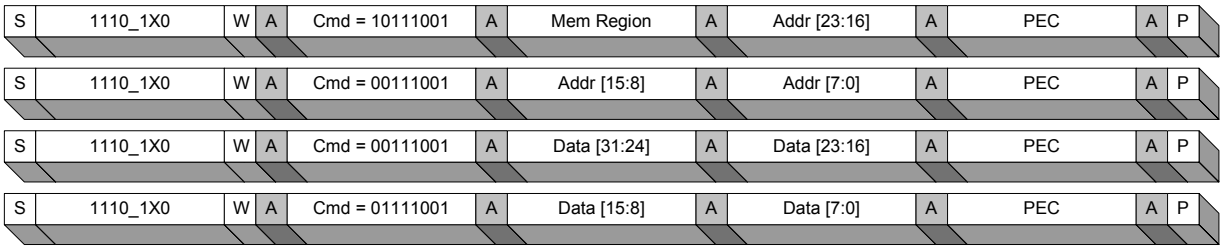


Figure 4-14. SMBus Word-Size Memory Register Write



4.12.3.7.3 SMBus Configuration and Memory Byte Writes

Figure 4-15. SMBus Configuration (Byte Write, PEC Enabled)

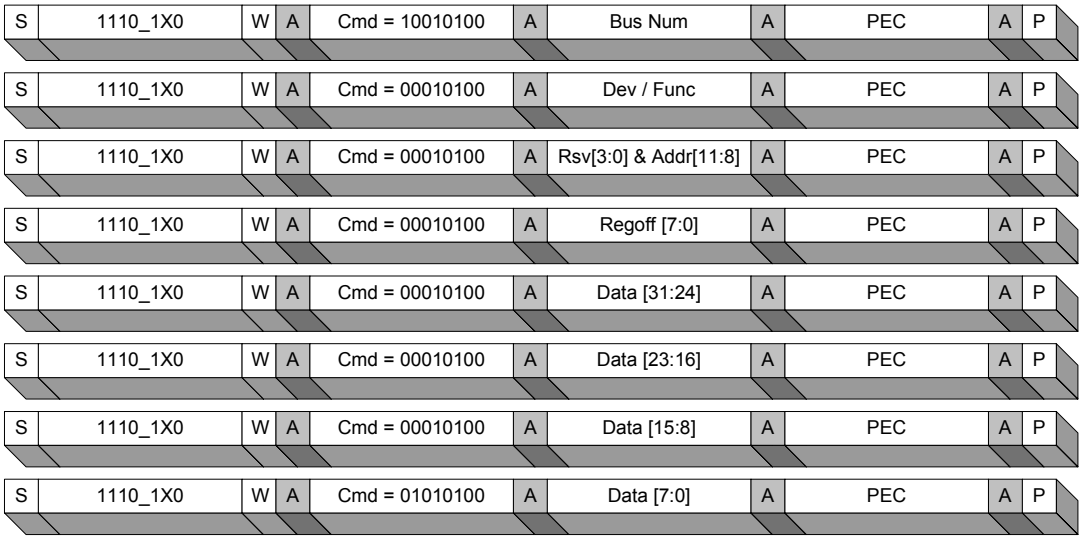
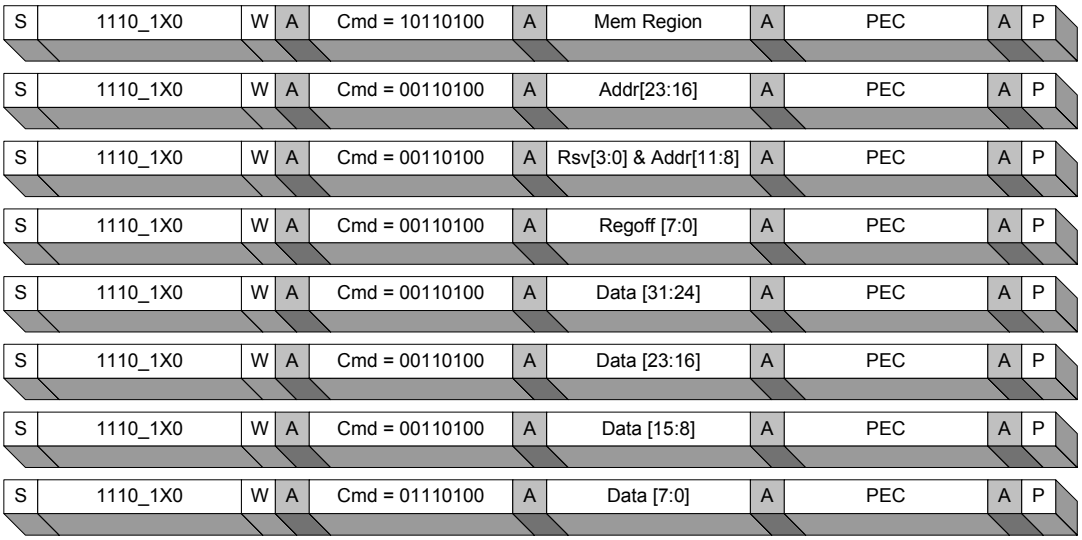


Figure 4-16. SMBus Memory (Byte Write, PEC Enabled)



4.12.4 GbE SMBus (Master/Slave)

See Chapter 22.0, "Management Interfaces" and Chapter 27.0, "GbE Platform Manageability" for the GbE SMBus interface description.

4.12.5 SMLINK1 Interface

SMLINK1 is used by an external controller to obtain system thermal data from sensors integrated into the components on the platform. See ***Section 4.19, "Thermal Management"*** for information about Thermal Reporting using SMLINK1.

4.13 Serial I/O Unit and Watchdog Timer (SIW) (B0:D31:F0)

4.13.1 Overview

The Serial I/O unit and Watchdog Timer (SIW) is similar to currently available Super I/O controllers. It is specifically designed to be integrated into the PCH and consists of two UARTs, a serial interrupt controller, a watchdog timer, and the LPC interface.

4.13.2 Features

LPC Interface Multiplexed Command, Address, and Data Bus

- 8-bit I/O transfers
- 16-bit address qualification for I/O transactions
- Serial IRQ interface compatible with serialized IRQ support for PCI systems

Serial Port

- Two full-function 16550-compatible serial ports
- Configurable I/O addresses and interrupts
- 16-byte FIFOs
- Supports up to 115 Kbps
- Programmable baud rate generator
- Modem control circuitry
- 14.7456 MHz and 48 MHz supported for UART baud clock input

Watchdog Timer (WDT)

Selectable Prescaler – about 1 MHz (1 µs to 1 s) and about 1 KHz (1 ms to 10 min)

- 33 MHz Clock (30 ns Clock Ticks)
- Multiple Modes (WDT and Free-Running)
- Free-Running Mode: One-stage timer - Toggles WDT_TOUT# after programmable time.
- WDT Mode: Two-stage timer (First stage generates interrupt, second stage drives WDT_TOUT# low)
  - First stage generates an SERIRQ interrupt (if enabled) after programmable time.
  - Second stage drives WDT_OUT# low or inverts the previous value.
  - Used only after first timeout occurs.
  - Status bit preserved in RTC well for possible error detection and correction.
  - Drives WDT_TOUT# if OUTPUT is enabled.
- Timer can be disabled (default state) or Locked (Hard reset required to disable WDT)
- WDT Automatic Reload of Preload value when WDT Reload Sequence is performed.

4.13.3 Functional Description

4.13.3.1 Host Processor Interface (LPC)

The host processor communicates with the SIW via the LPC bus. Access is through a series of read/ write registers and accomplished through I/O cycles. All registers are eight bits wide. The SIW registers include global configuration space and device specific regions accessed by setting the Logical Device Number in the SIW Configuration Register 07H (SCR7).

Table 4-48. Address Map

Address	Block Name	Logical Device
04Eh or 2Eh (SIU1_DTR# dependent)	Configuration Index	
04Fh or 2Fh (SIU1_DTR# dependent)	Configuration Data	
Base+(0-7)	Serial Port 1	04H
Base+(0-7)	Serial Port 2	05H
Base+(0-18)	Watchdog Timer	06H

See Section 7.1 for configuration register descriptions and information on setting the base address.

4.13.3.2 LPC Interface

The LPC interface is used to control all the logical blocks on the SIW. LPC bus signals use PCI 33 MHz electrical signal characteristics. See the *Low Pin Count (LPC) Interface Specification Rev 1.0.*

4.13.3.3 LPC Cycles

The following cycle types are supported by the LPC protocol.

Table 4-49. Supported LPC Cycle Types

Cycle Type	Transfer Size
I/O Write	1 Byte
I/O Read	1 Byte

The SIW ignores cycles that it does not support.

4.13.3.4 I/O Read and Write Cycles

The SIW is the target for I/O cycles. I/O cycles are initiated by the host for register or FIFO accesses and generally have minimal synchronization times.

Data transfers are assumed to be exactly 1-byte. If the CPU requested a 16-bit or 32-bit transfer, the host must divide it up into 8-bit transfers.

See the *LPC Interface Specification* for the sequence of cycles for the I/O Read and Write cycles.

4.13.3.5 Policy

The following rules govern the reset policy:

SIW_RESET# is tied to the internal PCI bus reset.

When SIW_RESET# goes active (low):

- The host drives the LFRAME# signal high, tri-states the LAD[3:0] signals.
- The SIU ignores LFRAME#, tri-states the LAD[3:0] pins.

Note: LPC bus signals from SIW are internally tied to the primary LPC interface of the PCH. Host LPC and SIW LPC names are used interchangeably throughout.

4.13.3.6 LPC Transfers

4.13.3.6.1 I/O Transfers

These are generally used for register or FIFO accesses, and generally have minimal synchronization times. The minimum number of wait-states between bytes is one. Data transfers are assumed to be exactly one byte. The host is responsible for breaking up larger data transfers into 8-bit cycles.

Table 4-50. I/O Sync Bits Description

Bits	Indication
0000	Synchronization achieved with no error.
0101	Indicates that synchronization not achieved yet, but the part is driving the bus.
0110	Indicates that synchronization not achieved yet, but the part is driving the bus and expects long synchronization
1010	Special Case: peripheral indicating errors.

4.13.4 LPC Logical Devices 4 and 5: Serial Ports (UART1 and UART2)

This section describes the Universal Asynchronous Receiver/Transmitter (UART) serial port used for the two UART integrated into the SIW. The UART can be controlled via programmed I/O. The basic programming model is the same for both UARTs with the only difference being the Logical Device Number assigned to each.

The serial port consists of a UART which supports all the functions of a standard 16550 UART including hardware flow control interface.

The UART performs serial-to-parallel conversion on data characters received from a peripheral device or a modem and parallel-to-serial conversion on data characters received from the processor. The processor can read the complete status of the UART at any time during the functional operation. Available status information includes the type and condition of the transfer operations being performed by the UART and any error conditions (parity, overrun, framing, or break interrupt).

The serial port can operate in either FIFO or non-FIFO mode. In FIFO mode, a 16-byte transmit FIFO holds data from the processor to be transmitted on the serial link and a 16-byte Receive FIFO buffers data from the serial link until read by the processor.

Each UART includes a programmable baud rate generator which is capable of dividing the baud clock input by divisors of one to (2^{16} -1) and producing a 16X clock to drive the internal transmitter and receiver logic. Each UART has complete modem control capability and a processor interrupt system. Interrupts can be programmed to the

user's requirements, minimizing the computing required to handle the communications link. Each UART can operate in a polled or an interrupt driven environment as configured by software.

The baud rate generator input is a function of the UART_CLK and a configurable pre-divide of 1, 8, or 26. See SIW Configuration (address 29h) in Section 15.4.1, "SIW Configuration Register Summary" on page 746. The output of the baud rate generator is 16 times the baud rate.

Table 4-51. UART Clock Divider Support

Clock Frequency	1.8432 MHz	14.7456 MHz	48.0 MHz
Predivide Value	1	8	26
Generator Frequency	1.8432 MHz	1.8432 MHz	1.8462 MHz

Table 4-52. Baud Rate Example

Desired Baud Rate	Divisor	% error @ 1.8432	% error @ 1.8462*
300	384		0.16
1200	96		0.16
2400	48		0.16
4800	24		0.16
9600	12		0.16
19200	6		0.16
38400	3		0.16
56000	2	2.8	3
115200	1		0.16

4.13.4.1 UART Feature List

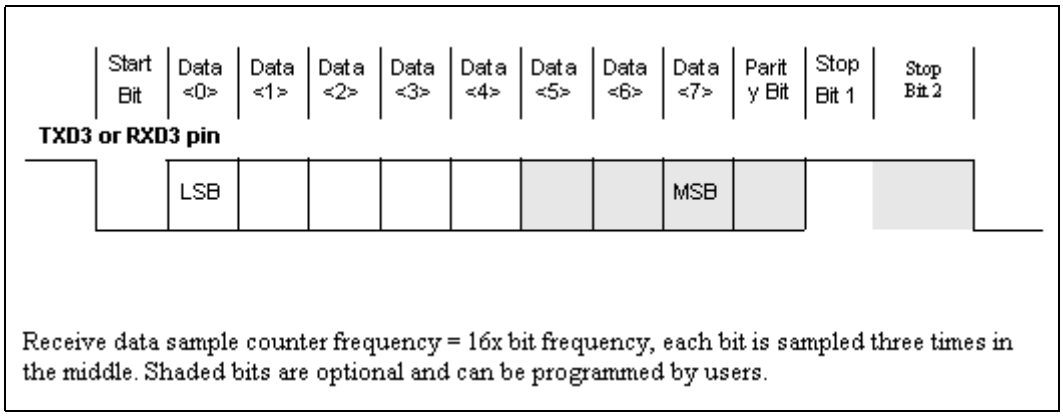
- Functionally compatible with National Semiconductor's PC16550D
- Adds or deletes standard asynchronous communications bits (start, stop, and parity) to or from the serial data
- Independently controlled transmit, receive, line status and data set interrupts
- Programmable baud rate generator allows division of clock by 1 to (216 -1) and generates an internal 16X clock
- Modem control functions (CTS#, RTS#, DSR#, DTR#, RI#, and DCD#)
- Fully programmable serial-interface characteristics:
- 5, 6, 7, or 8-bit characters
- Even, odd, or no parity detection
- 1, 1-1/2, or 2 stop bit generation
- Baud rate generation (up to 115 Kbps)
- False start bit detection
- 16-byte Receive FIFO
- Complete status reporting capability

- Line break generation and detection
- Internal diagnostic capabilities include:
  - — Loopback controls for communications link fault isolation
  - — Break, parity, overrun, and framing error simulation
  - — Fully prioritized interrupt system controls

4.13.4.2 UART Operational Description

The format of a UART data frame is shown in Figure 4-17.

Figure 4-17. Example UART Data Frame



Each data frame is between seven bits and 12 bits long depending on the size of data programmed, if parity is enabled and if two stop bits is selected. The frame begins with a start bit that is represented by a high to low transition. Next, 5 to 8 bits of data are transmitted, beginning with the least significant bit. An optional parity bit follows, which is set if even parity is enabled and an odd number of ones exist within the data byte, or if odd parity is enabled and the data byte contains an even number of ones. The data frame ends with one, one and a half or two stop bits as programmed by the user, which is represented by one or two successive bit periods of a logic one.

The unit is disabled upon reset, the user needs to enable the unit by setting bit six of Interrupt Enable Register. When the unit is enabled, the receiver starts looking for the start bit of a frame; the transmitter starts transmitting data to the transmit data pin if there is data available in the transmit FIFO. Transmit data can be written to the FIFO before the unit is enabled. When the unit is disabled, the transmitter/receiver finishes the current byte being transmitted/received if it is in the middle of transmitting/receiving a byte and stops transmitting/receiving more data.

An SIU_RESET# to the SIU forces the internal register and output signals on the serial port to the values listed in Table 4-53.

Table 4-53. UART Register/Signal Reset States

Register/Signal	Reset Control	Reset State
Interrupt Enable Register	RESET	All bits are low.
Interrupt ID Register	RESET	Bit 0 is forced high. Bits 1-3 and 6-7 are forced low. Bits 4-5 are permanently low.
Line Control Register	RESET	All bits are forced low.

Table 4-53. UART Register/Signal Reset States

Register/Signal	Reset Control	Reset State
Line Status Register	RESET	Bits 0-4, 7 are forced low. Bits 5 and 6 are forced high.
Modem Control Register	RESET	Bits 0, 1, 2, 3, 4 are forced low. Bits 5, 6, 7 are permanently low.
Modem Status Register	RESET/Modem signal, read MSR for bits 3-0.	Low
Infrared Selection Register	RESET	All bits are permanently low.
Txd	RESET	High
Int	RESET/ clear LINE STATUS REG	Low
rts_n	RESET	High
dtr_n	RESET	High

4.13.4.3 Programmable Baud Rate Generator

The UART contains a programmable Baud Rate Generator that is capable of taking the UART_CLK input and dividing it by any divisor from 1 to (2^{16} -1). The output frequency of the Baud Rate Generator is 16 times the baud rate. Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor Latches must be loaded during initialization to ensure proper operation of the Baud Rate Generator. If both Divisor Latches are loaded with 0, the 16X output clock is stopped. Upon loading either of the Divisor latches, a 16-bit baud counter is immediately loaded. This prevents long counts on initial load. Access to the Divisor latch can be done with a word write.

Note: The UART_CLK is the SIW_CLK input divided by the prescaler set by the SIW Configuration Register (Offset 29h).

The baud rate of the data shifted in/out of the UART is given by:

Baud Rate = UART_CLK(MHz)/[16X Divisor]

For example, if UART_CLK is 14.7456 MHz and the divisor is 96, the baud rate is 9600.

A divisor value of 0 in the Divisor Latch Register is not allowed. The reset value of the divisor is 02.

4.13.4.4 FIFO Operation

4.13.4.4.1 FIFO Interrupt Mode Operation

4.13.4.4.2 Receiver Interrupt

When the Receive FIFO and receiver interrupts are enabled (FCR[0]=1 and IER[0]=1), receiver interrupts occur as follows:

- The receive data available interrupt is invoked when the FIFO has reached its programmed trigger level. The interrupt is cleared when the FIFO drops below the programmed trigger level.
- The IIR receive data available indication also occurs when the FIFO trigger level is reached, and like the interrupt, the bits are cleared when the FIFO drops below the trigger level.

- The receiver line status interrupt (IIR = C6H), as before, has the highest priority. The receiver data available interrupt (IIR=C4H) is lower. The line status interrupt occurs only when the character at the top of the FIFO has errors.
- The data ready bit (DR in LSR register) is set to 1 as soon as a character is transferred from the shift register to the Receive FIFO. This bit is reset to 0 when the FIFO is empty.

4.13.4.4.3 Character Timeout Interrupt

When the receiver FIFO and receiver time out interrupt are enabled, a character timeout interrupt occurs when all of the following conditions exist:

- At least one character is in the FIFO.
- The last received character was longer than four continuous character times ago (if two stop bits are programmed the second one is included in this time delay).
- The most recent processor read of the FIFO was longer than four continuous character times ago.
- The receive FIFO trigger level is greater than one.

The maximum time between a received character and a timeout interrupt is 160 ms at 300 baud with a 12-bit receive character (for example, one start, eight data, one parity, and two stop bits).

When a time out interrupt occurs, it is cleared and the timer is reset when the processor reads one character from the receiver FIFO. If a timeout interrupt has not occurred, the timeout timer is reset after a new character is received or after the processor reads the receiver FIFO.

4.13.4.4.4 Transmit Interrupt

When the transmitter FIFO and transmitter interrupt are enabled (FCR[0]=1, IER[1]=1), transmit interrupts occur as follows:

The transmitter holding register interrupt occurs when the transmit FIFO is empty; it is cleared as soon as the transmitter holder register is written to (1 to 16 characters may be written to the transmit FIFO while servicing this interrupt) or the IIR is read.

The transmitter FIFO empty indications are delayed one character time minus the last stop bit time whenever the following occurs: THRE = 1 and there have not been at least two bytes at the same time in the transmit FIFO since the last THRE = 1. The first transmitter interrupt after changing FCRO is immediate if it is enabled.

4.13.4.5 FIFO Polled Mode Operation

With the FIFOs enabled (TRFIFOE bit of FCR set to 1), setting IER[3:0] to all zeros puts the serial port in the FIFO polled mode of operation. Since the receiver and the transmitter are controlled separately, either one or both can be in the polled mode of operation. In this mode, software checks receiver and transmitter status via the LSR. As stated in the register description:

- LSR[0] is set as long as there is one byte in the receiver FIFO.
- LSR[1] through LSR[4] specify which error(s) has occurred for the character at the top of the FIFO. Character error status is handled the same way as interrupt mode. The IIR is not affected since IER[2] = 0.
- LSR[5] indicates when the transmitter FIFO needs data.
- LSR[6] indicates that both the transmitter FIFO and shift register are empty.
- LSR[7] indicates whether there are any errors in the receiver FIFO.

4.13.5 LPC Logical Device 6: Watchdog Timer

4.13.5.1 Overview

This device is a Watchdog timer that provides a resolution that ranges from 1 µs to 10 minutes. The timer uses a 35-bit down-counter.

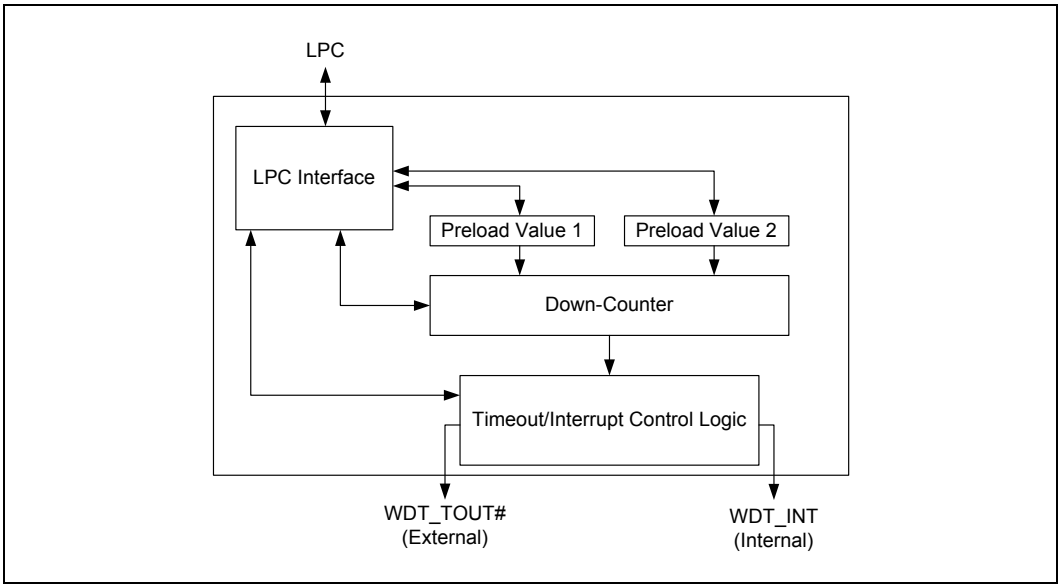
The counter is loaded with the value from the first Preload register. The timer is then enabled and it starts counting down. The time at which the WDT first starts counting down is called the first stage. If the host fails to reload the WDT before the 35-bit down counter reaches zero the WDT generates an internal interrupt.

After the interrupt is generated the WDT loads the value from the second Preload register into the WDT's 35-bit Down-Counter and starts counting down. The WDT is now in the second stage. If the host still fails to reload the WDT before the second timeout, the WDT drives the WDT_TOUT# pin low and sets the timeout bit (WDT_TIMEOUT). This bit indicates that the system has become unstable. The WDT_TOUT# pin is held low until the system is reset or the WDT times out again (Depends on TOUT_CNF). The process of reloading the WDT involves the following sequence of writes:

1. Write "80" to offset BAR1 + 0Ch
2. Write "86" to offset BAR1 + 0Ch
3. Write '1' to WDT_RELOAD in Reload Register.

The same process is used for setting the values in the preload registers. The only difference exists in step 3. Instead of writing a '1' to the WDT_RELOAD, you write the desired preload value into the corresponding Preload register. This value is not loaded into the 35-bit down counter until the next time the WDT reenters the stage. For example, if Preload Value 2 is changed, it is not loaded into the 35-bit down counter until the next time the WDT enters the second stage.

Figure 4-18. WDT Block Diagram



4.13.5.2 Theory Of Operation

4.13.5.2.1 RTC Well and WDT_TOUT# Functionality

The WDT_TIMEOUT bit is set to a '1' when the WDT 35-bit down counter reaches zero for the second time in a row. Then the WDT_TOUT# pin is toggled LOW by the WDT from the ILB. The board designer must attach the WDT_TOUT# to the appropriate external signal. If WDT_TOUT_CNF is a '1' the WDT toggles WDT_TOUT# again the next time a time out occurs. Otherwise WDT_TOUT# is driven low until the system is reset or power is cycled.

4.13.5.2.2 Register Unlocking Sequence

The register unlocking sequence is necessary whenever writing to the RELOAD register or either PRELOAD_VALUE registers. The host must write a sequence of two writes to offset BAR1 + 0Ch before attempting to write to either the WDT_RELOAD and WDT_TIMEOUT bits of the RELOAD register or the PRELOAD_VALUE registers. The first writes are "80" and "86" (in that order) to offset BAR1 + 0Ch. The next write is to the proper memory mapped register (for example, RELOAD, PRELOAD_VALUE_1, PRELOAD_VALUE_2). Any deviation from the sequence (writes to memory-mapped registers) causes the host to have to restart the sequence.

When performing register unlocking, software must issue the cycles using byte access only. Otherwise the unlocking sequence does not work properly.

The following is an example of how to prevent a timeout:

1. Write "80" to offset BAR1 + 0Ch
2. Write "86" to offset BAR1 + 0Ch
3. Write a '1' to RELOAD [8] (WDT_RELOAD) of the Reload Register

Note: Any subsequent writes require that this sequence be performed again.

4.13.5.2.3 Reload Sequence

To keep the timer from causing an interrupt or driving WDT_TOUT#, the timer must be updated periodically. Other timers refer to "updating the timer" as "kicking the timer". The frequency of updates required is dependent on the value of the Preload values. To update the timer the Register Unlocking Sequence must be performed followed by writing a '1' to bit 8 at offset BAR1 + 0Ch within the watchdog timer memory mapped space. This sequence of events is referred to as the "Reload Sequence".

4.13.5.2.4 Low Power State

The Watchdog Timer does not operate when PCICLK is stopped.

4.14 General Purpose I/O (B0:D31:F0)

The system contains up to 67 General Purpose Input/Output (GPIO) signals. Each GPIO can be configured as an input or output signal. The number of inputs and outputs varies depending on the configuration. The following list is a brief summary of GPIO features:

- Capability to mask Suspend well GPIOs from CF9h events configured via GP_RST_SEL registers)
- Added capability to program GPIO prior to switching to output

4.14.1 Power Wells

Some GPIOs exist in the suspend power plane. Care must be taken to make sure GPIO signals are not driven high into powered-down planes. Some GPIOs may be connected to pins on devices that exist in the core well. If these GPIOs are outputs, there is a danger that a loss of core power (PWROK low) or a Power Button Override event results in the system driving a pin to a logic 1 to another device that is powered down.

4.14.2 SMI# SCI and NMI Routing

The routing bits for GPIO[15:0] allow an input to be routed to SMI#, SCI, NMI or neither. A bit can be routed to either an SMI# or an SCI, but not both.

4.14.3 Triggering

GPIO[15:0] have "sticky" bits on the input. See the GPE0_STS register and the ALT_GPI_SMI_STS register. As long as the signal goes active for at least 2 clock cycles, the system keeps the sticky status bit active. The active level can be selected in the GP_INV register. This does not apply to GPI_NMI_STS residing in GPIO IO space.

If the system is in an S0 or an S1 state, the GPI inputs are sampled at 33 MHz, so the signal only needs to be active for about 60 ns to be latched. In the S3–S5 states, the GPI inputs are sampled at 32.768 kHz, and thus must be active for at least 61 microseconds to be latched. GPIs that are in the core well are not capable of waking the system from sleep states where the core well is not powered.

If the input signal is still active when the latch is cleared, it sets again. Another edge trigger is not required. This makes these signals "level" triggered inputs.

4.14.4 GPIO Registers Lockdown

The following GPIO registers are locked down when the GPIO Lockdown Enable (GLE) bit is set. The GLE bit resides in B0:D31:F0:GPIO Control (GC) register.

- Offset 00h: GPIO_USE_SEL[31:0]
- Offset 04h: GP_IO_SEL[31:0]
- Offset 0Ch: GP_LVL[31:0]
- Offset 28h: GPI_NMI_EN[15:0]
- Offset 2Ch: GPI_INV[31:0]
- Offset 30h: GPIO_USE_SEL2[63:32]
- Offset 34h: GPI_IO_SEL2[63:32]
- Offset 38h: GP_LVL2[63:32]
- Offset 40h: GPIO_USE_SEL3[95:64]
- Offset 44h: GPI_IO_SEL3[95:64]
- Offset 48h: GP_LVL3[95:64]
- Offset 60h: GP_RST_SEL[31:0]
- Offset 64h: GP_RST_SEL2[63:32]
- Offset 68h: GP_RST_SEL3[95:64]

Once these registers are locked down, they become Read-Only registers and any software writes to these registers has no effect. To unlock the registers, the GPIO Lockdown Enable (GLE) bit is required to be cleared to '0'. When the GLE bit changes from a '1' to a '0' a System Management Interrupt (SMI#) is generated if enabled.

Once the GPIO_UNLOCK_SMI bit is set, it can not be changed until a PLTRST# occurs. This ensures that only BIOS can change the GPIO configuration. If the GLE bit is cleared by unauthorized software, BIOS will set the GLE bit again when the SMI# is triggered and these registers will continue to be locked down.

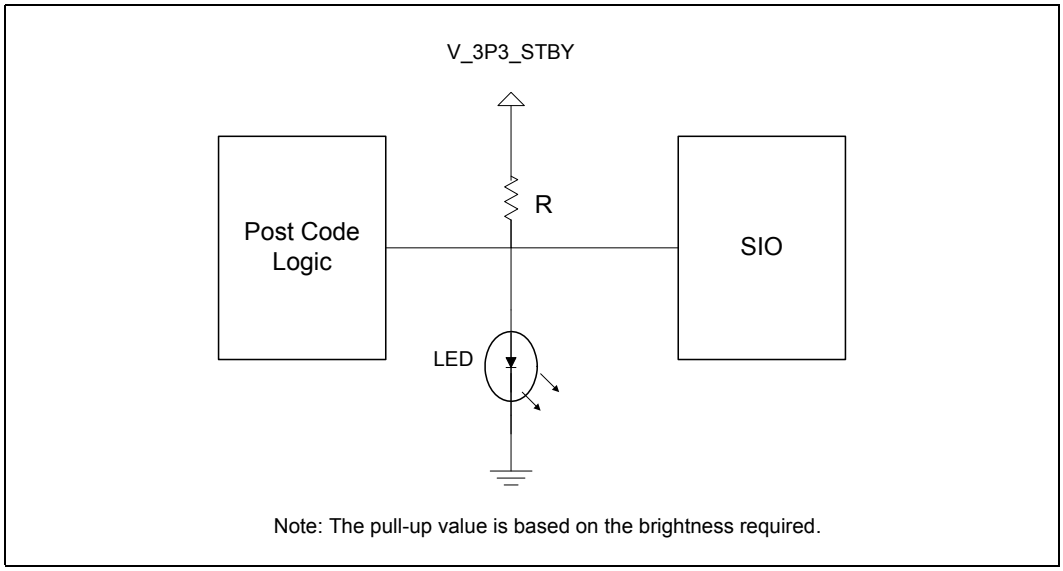
4.14.5 Serial POST Codes Over GPIO

The system provides an extended capability which allows system software to serialize POST or other messages on GPIO. This capability negates the requirement for dedicated diagnostic LEDs on the platform. Additionally, based on the newer BTX form factors, the PCI bus as a target for POST codes is increasingly difficult to support as the total number of PCI devices supported are decreasing.

4.14.5.1 Theory of Operation

The POST code serialization logic is shared with GPIO. These GPIOs will likely be shared with LED control offered by the Super I/O (SIO) component. Figure 4-19 shows a likely configuration.

Figure 4-19. Serial Post over GPIO Reference Circuit



The anticipated usage model is that either the system or the SIO can drive a pin low to turn off an LED. In the case of the power LED, the SIO would normally leave its corresponding pin in a high-Z state to allow the LED to turn on. In this state, the system can blink the LED by driving its corresponding pin low and subsequently tri-stating the buffer. The I/O buffer should not drive a '1' when configured for this functionality and should be capable of sinking 24mA of current.

An external optical sensing device can detect the on/off state of the LED. By externally post-processing the information from the optical device, the serial bit stream can be recovered. The hardware will supply a 'sync' byte before the actual data transmission to allow external detection of the transmit frequency. The frequency of transmission should be limited to 1 transition every 1µs to ensure the detector can reliably sample the on/off state of the LED. To allow flexibility in pull-up resistor values for power optimization, the frequency of the transmission is programmable via the DRS field in the GP_GB_CMDSTS register.

The serial bit stream is Manchester encoded. This choice of transmission ensures that a transition is seen on every clock. The 1 or 0 data is based on the transmission happening during the high or low phase of the clock.

As the clock is encoded within the data stream, hardware must ensure that the Z-0 and 0-Z transitions are glitch-free. Driving the pin directly from a flop or through glitch-free logic are possible methods to meet the glitch-free requirement.

A simplified hardware/software register interface provides control and status information to track the activity of this block. Software enabling the serial blink capability should implement an algorithm referenced in the following list to send the serialized message on the enabled GPIO:

1. Read the Go/Busy status bit in the GP_GB_CMDSTS register and verify it is cleared. This will ensure that the GPIO is idled and a previously requested message is still not in progress.
2. Write the data to serialize into the GP_GB_DATA register.
3. Write the DLS and DRS values into the GP_GB_CMDSTS register and set the Go bit. This may be accomplished using a single write.

The reference diagram shows the LEDs being powered from the suspend supply. By providing a generic capability that can be used both in the main and the suspend power planes maximum flexibility can be achieved. A key point to make is that the system does not unintentionally drive the LED control pin low unless a serialization is in progress. System board connections utilizing this serialization capability are required to use the same power plane controlling the LED as system GPIO pin. Otherwise, the system GPIO may float low during the message and prevent the LED from being controlled from the SIO. The hardware will only be serializing messages when the core power well is powered and the processor is operational.

Care should be taken to prevent the system from driving an active '1' on a pin sharing the serial LED capability. Since the SIO could be driving the line to 0, having the system drive a 1 would create a high current path. A recommendation to avoid this condition involves choosing a GPIO defaulting to an input. The GP_SER_BLINK register should be set first before changing the direction of the pin to an output. This sequence ensures the open-drain capability of the buffer is properly configured before enabling the pin as an output.

4.14.5.2 Serial Message Format

In order to serialize the data onto the GPIO, an initial state of high-Z is assumed. The SIO is required to have its LED control pin in a high-Z state as well to allow the system to blink the LED (see the reference diagram).

The three components of the serial message include the sync, data, and idle fields. The sync field is 7 bits of '1' data followed by 1 bit of '0' data. Starting from the high-Z state (LED on) provides external hardware a known initial condition and a known pattern. In case one or more of the leading 1 sync bits are lost, the 1s followed by 0 provide a clear indication of 'end of sync'. This pattern is used to 'lock' external sampling logic to the encoded clock.

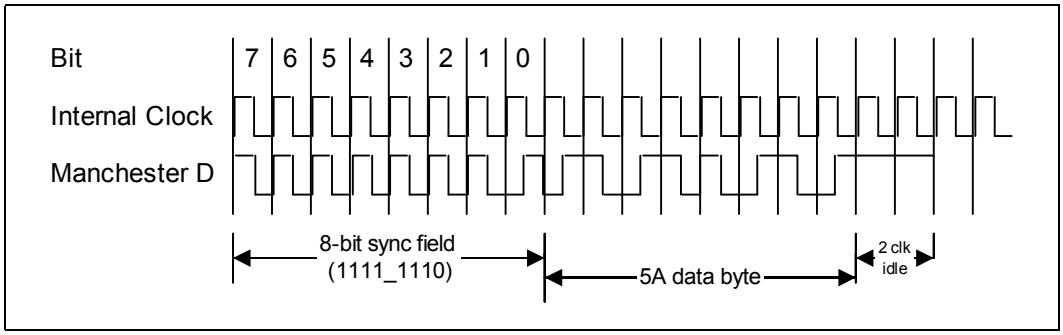
The data field is shifted out with the highest byte first (MSB). Within each byte, the most significant bit is shifted first (MSb).

The idle field is enforced by the hardware and is at least 2 bit times long. The hardware does not clear the Busy and Go bits until this idle time is met. Supporting the idle time in hardware prevents time-based counting in BIOS as the hardware is immediately ready for the next serial code when the Go bit is cleared. The idle state is represented as a high-Z condition on the pin. If the last transmitted bit is a 1, returning to the idle

state results in a final 0-1 transition on the output Manchester data. Two full bit times of idle correspond to a count of 4 time intervals (the width of the time interval is controlled by the DRS field).

The waveform in Figure 4-20 shows a 1-byte serial write with a data byte of 5Ah. The internal clock and bit position are for reference purposes only. The Manchester D is the resultant data generated and serialized onto the GPIO. Since the buffer is operating in open-drain mode, the transitions are from high-Z to 0 and back.

Figure 4-20. 1-byte Serial Write with a Data Byte of 5Ah



4.15 SATA* Host Controller (B0:D31:F2 & B0:D31:F5)

The SATA* Interface contains two controllers (B0:D31:F2 and B0:D31:F5) to support AHCI/IDE modes of operation for different operating systems. Each controller is configured to support two physical ports. These are physically numbered as Port 4 and Port 5.

4.15.1 SATA* Ports 4 and 5 Numbering

The SATA* Interface is adopted from a legacy SATA* Interface which supported six interface ports (0 through 5) in both the IDE and AHCI modes.

In the legacy interface usage, the controllers were used to support 6 ports as follows:

- In AHCI mode, controller #1 supported all 6 ports (0 through 5).
- In Native IDE mode, controller #1 supported 4 ports (0 through 3) and controller #2 supported 2 ports 4 and 5

In this product, two controllers are used to support 2 ports as follows:

- In AHCI mode, controller #1 (B0:D31:F2) supports 2 ports 4 and 5.
- In Native IDE mode, controller #2 (B0:D31:F5) supports 2 ports 4 and 5.

4.15.2 SATA* Feature Support

Feature	Description	System Support
Native Command Queuing (NCQ)	Allows the device to reorder commands for more efficient data transfers	Supported
Auto Activate for DMA	Collapses a DMA Setup then DMA Activate sequence into a DMA Setup only	Supported
Hot Plug Support	Allows for device detection without power being applied and ability to connect and disconnect devices without prior notification to the system	Supported
Asynchronous Signal Recovery	Provides a recovery from a loss of signal or establishing communication after hot plug	Supported
3 Gb/s Transfer Rate	Capable of data transfers up to 3Gb/s	Supported
ATAPI Asynchronous Notification	A mechanism for a device to send a notification to the host that the device requires attention	Supported
Host & Link Initiated Power Management	Capability for the host controller or device to request Partial and Slumber interface power states	Supported
Staggered Spin-Up	Enables the host the ability to spin up hard drives sequentially to prevent power load problems on boot	Supported
Legacy Compatibility Mode	Compatibility with Legacy Modes	Not Supported
Command Completion Coalescing	Reduces interrupt and completion overhead by allowing a specified number of commands to complete and then generating an interrupt to process the commands	Not Supported
Port Multiplier	A mechanism for one active host connection to communicate with multiple devices	Not Supported
External SATA*	Technology that allows for an outside the box connection of up to 2 meters (when using the cable defined in SATA-IO)	Not Supported

4.15.3 Theory of Operation

4.15.3.1 Standard ATA Emulation

The system contains a set of registers that shadow the contents of the legacy IDE registers. The behavior of the Command and Control Block registers, PIO, and DMA data transfers, resets, and interrupts are all emulated.

Note: The system asserts INTR when the master device completes the EDD command regardless of the command completion status of the slave device. If the master completes EDD first, an INTR is generated and BSY remains '1' until the slave completes the command. If the slave completes EDD first, BSY is '0' when the master completes the EDD command and asserts INTR. Software must wait for busy to clear (0) before completing an EDD command, as required by the ATA5 through ATA7 (T13) industry standards.

4.15.3.2 48-Bit LBA Operation

The SATA* host controller supports 48-bit LBA through the host-to-device register FIS when accesses are performed via writes to the task file. The SATA* host controller ensures that the correct data is put into the correct byte of the host-to-device FIS.

There are special considerations when reading from the task file to support 48-bit LBA operation. Software may need to read all 16-bits. Since the registers are only 8-bits wide and act as a FIFO, a bit must be set in the device/control register, which is at offset 3F6h for primary and 376h for secondary (or their native counterparts).

If software clears bit 7 of the control register before performing a read, the last item written is returned from the FIFO. If software sets bit 7 of the control register before performing a read, the first item written is returned from the FIFO.

4.15.4 SATA* Swap Bay Support

The system provides for basic SATA* swap bay support using the PSC register configuration bits and power management flows. A device can be powered down by software and the port can then be disabled, allowing removal and insertion of a new device.

Note: This SATA* swap bay operation requires board hardware (implementation specific), BIOS, and operating system support.

4.15.5 Hot Plug Operation

The system supports Hot Plug Surprise removal and Insertion Notification in the PARTIAL, SLUMBER and Listen Mode states when used with Low Power Device Presence Detection. Software can take advantage of power savings in the low power states while enabling hot plug operation. See the AHCI specification for details.

4.15.5.1 Low Power Device Presence Detection

Low Power Device Presence Detection enables SATA* Link Power Management to co-exist with hot plug (insertion and removal) without interlock switch or cold presence detect. The detection mechanism allows Hot Plug events to be detectable by hardware across all link power states (Active, PARTIAL, SLUMBER) as well as AHCI Listen Mode.

If the Low Power Device Presence Detection circuit is disabled, the system reverts to Hot Plug Surprise Removal Notification (without an interlock switch) mode that is mutually exclusive of the PARTIAL and SLUMBER power management states.

4.15.6 Power Management Operation

Power management of SATA* controller and ports covers operations of the host controller and the SATA* wire.

4.15.6.1 Power State Mappings

The D0 PCI power management state for device is supported by SATA* controller.

SATA* devices may also have multiple power states. From parallel ATA, three device states are supported through ACPI:

- D0 – Device is working and instantly available.
- D1 – device enters when it receives a STANDBY IMMEDIATE command. Exit latency from this state is in seconds
- D3 – from the SATA* device's perspective, no different than a D1 state, in that it is entered via the STANDBY IMMEDIATE command. However, an ACPI method is also called which resets the device and then cut its power.

Each of these device states are subsets of the host controller's D0 state.

SATA* defines three PHY layer power states, which have no equivalent mappings to parallel ATA. They are:

- PHY READY – PHY logic and PLL are both on and active
- Partial – PHY logic is powered, but in a reduced state. Exit latency is no longer than 10 ns
- Slumber – PHY logic is powered, but in a reduced state. Exit latency can be up to 10 ms.

Since these states have much lower exit latency than the ACPI D1 and D3 states, the SATA* controller defines these states as sub-states of the device D0 state.

4.15.6.2 Power State Transitions

4.15.6.2.1 Partial and Slumber State Entry/Exit

The partial and slumber states save interface power when the interface is idle. It would be most analogous to PCI CLKRUN# (in power savings, not in mechanism), where the interface can have power saved while no commands are pending. The SATA* controller defines PHY layer power management (as performed via primitives) as a driver operation from the host side, and a device proprietary mechanism on the device side. The SATA* controller accepts device transition types, but does not issue any transitions as a host. All received requests from a SATA* device is ACKed.

When an operation is performed to the SATA* controller such that it needs to use the SATA* cable, the controller must check whether the link is in the Partial or Slumber states, and if so, must issue a COM_WAKE to bring the link back online. Similarly, the SATA* device must perform the same action.

4.15.6.2.2 Device D1, D3 States

These states are entered after some period of time when software has determined that no commands is sent to this device for some time. The mechanism for putting a device in these states does not involve any work on the host controller, other then sending commands over the interface to the device. The command most likely to be used in ATA/ATAPI is the “STANDBY IMMEDIATE” command.

4.15.6.2.3 Host Controller $D3_{HOT}$ State

After the interface and device have been put into a low power state, the SATA* host controller may be put into a low power state. This is performed via the PCI power management registers in configuration space. There are two important aspects to note when using PCI power management:

1. When the power state is D3, only accesses to configuration space are allowed. Any attempt to access the memory or I/O spaces results in master abort.
2. When the power state is D3, no interrupts may be generated, even if they are enabled. If an interrupt status bit is pending when the controller transitions to D0, an interrupt may be generated.

When the controller is put into D3, it is assumed that software has properly shut down the device and disabled the ports. Therefore, there is no need to sustain any values on the port wires. The interface is treated as if no device is present on the cable, and power is minimized.

When returning from a D3 state, an internal reset is not performed.

4.15.6.2.4 Non-AHCI Mode PME# Generation

When in non-AHCI mode (legacy mode) of operation, the SATA* controller does not generate PME#. This includes attach events (since the port must be disabled), or interlock switch events (via the SATAGP pins).

4.15.6.3 SMI Trapping (APM)

Bus0:Device 31:Function2:Offset C0h contains control for generating SMI# on accesses to the IDE I/O spaces. These bits map to the legacy ranges (1F0–1F7h, 3F6h, 170–177h, and 376h) and native IDE ranges defined by PCMDBA, PCTLBA, SCMDBA an SCTLBA. If the SATA* controller is in legacy mode and is using these addresses, accesses to one of these ranges with the appropriate bit set causes the cycle to not be forwarded to the SATA* controller, and for an SMI# to be generated. If an access to the Bus-Master IDE registers occurs while trapping is enabled for the device being accessed, then the register is updated, an SMI# is generated, and the device activity status bits are updated indicating that a trap occurred.

4.15.7 SATALED#

The SATALED# output is driven whenever the BSY bit is set in any SATA* port. The SATALED# is an active-low open-drain output. When SATALED# is low, the LED should be active. When SATALED# is high, the LED should be inactive.

4.15.8 AHCI Operation

The system provides hardware support for Advanced Host Controller Interface (AHCI), a programming interface for SATA* host controllers developed through a joint industry effort. AHCI defines transactions between the SATA* controller and software and enables advanced performance and usability with SATA*. Platforms supporting AHCI may take advantage of performance features such as no master/slave designation for SATA* devices—each device is treated as a master—and hardware assisted native command queuing. AHCI also provides usability enhancements such as hot-plug. AHCI requires appropriate software support (for example, an AHCI driver) and for some features, hardware support in the SATA* device or additional platform hardware.

The system supports all of the mandatory features of the *Serial ATA Advanced Host Controller Interface Specification*, Revision 1.2 and many optional features, such as hardware assisted native command queuing, aggressive power management, LED indicator support, and hot-plug through the use of interlock switch support (additional platform hardware and software may be required depending upon the implementation).

Note: For reliable device removal notification while in AHCI operation without the use of interlock switches (surprise removal), interface power management should be disabled for the associated port. See Section 7.3.1 of the *AHCI Specification* for more information.

4.15.9 SGPIO Signals

The SGPIO signals, in accordance to the SFF-8485 specification, support per-port LED signaling. These signals are not related to SATALED#, which allows for simplified indication of SATA* command activity. The SGPIO group interfaces with an external controller chip that fetches and serializes the data for driving across the SGPIO bus. The output signals then control the LEDs. This feature is only valid in AHCI mode.

4.15.9.1 Mechanism

The enclosure management for SATA* Controller 1 (Bus0:Device 31: Function 2) involves sending messages that control LEDs in the enclosure. The messages for this function are stored after the normal registers in the AHCI BAR, at Offset 580h bytes from the beginning of the AHCI BAR as specified by the EM_LOC global register.

Software creates messages for transmission in the enclosure management message buffer. The data in the message buffer should not be changed if CTL.TM bit is set by software to transmit an update message. Software should only update the message buffer when CTL.TM bit is cleared by hardware otherwise the message transmitted is indeterminate. Software then writes a register to cause hardware to transmit the message or take appropriate action based on the message content. The software should only create message types supported by the controller, which is LED messages. If the software creates other non LED message types (for example, SAF-TE, SES-2), the SGPIO interface may hang and the result is indeterminate.

During reset all SGPIO pins are in tri-state. The interface will continue to be in tri-state after reset until the first transmission occurs when software programs the message buffer and sets the transmit bit CTL.TM. The SATA* Host controller will initiate the transmission by driving SCLOCK and at the same time drive the SLOAD to ‘0’ prior to the actual bit stream transmission. The Host will drive SLOAD low for at least 5 SCLOCK then only start the bit stream by driving the SLOAD to high. SLOAD is driven high for 1 SCLOCK follow by vendor specific pattern that is default to “0000” if software has yet to program the value. A total of 21-bit stream from 7 ports (Port0, Port1, Port2, Port3, Port4 Port5 and Port6) of 3-bit per port LED message is transmitted on SDATAOUT0 pin after the SLOAD is driven high for 1 SCLOCK. Only 3 ports (Port4, Port5 and Port6) of 9 bit total LED message follow by 12 bits of tri-state value is transmitted out on SDATAOUT1 pin.

All the default LED message values is high prior to software setting them, except the Activity LED message that is configured to be hardware driven that is generated based on the activity from the respective port. All the LED message values is driven to ‘1’ for the port that is un-implemented as indicated in the Port Implemented register regardless of the software programmed value through the message buffer.

There are two ways to reset the system’s SGPIO interface: asynchronous reset and synchronous reset. Asynchronous reset is caused by platform reset to cause the SGPIO interface to be tri-state asynchronously. Synchronous reset is caused by setting the CTL.RESET bit, clearing the GHC.AE bit or HBA reset, where Host Controller completes the existing full bit stream transmission then only tri-state all the SGPIO pins. After the reset, both synchronous and asynchronous, the SGPIO pins stays tri-stated.

Note: The Host Controller does not guarantee to cause the target SGPIO device or controller to be reset. Software is responsible to keep the SGPIO interface in tri-stated state for 2 seconds to cause a reset on the target of the SGPIO interface.

4.15.9.2 Message Format

Messages are constructed with a one Dword header that describes the message to be sent followed by the actual message contents. The first Dword is constructed as follows:

Bit	Description
31:28	Reserved
27:24	Message Type (MTYPE): Specifies the type of the message. The message types are: 0h = LED 1h = SAF-TE 2h = SES-2 3h = SGPIO (register based interface) All other values reserved
23:16	Data Size (DSIZE): Specifies the data size in bytes. If the message (enclosure services command) has a data buffer that is associated with it that is transferred, the size of that data buffer is specified in this field. If there is no separate data buffer, this field has have a value of '0'. The data directly follows the message in the message buffer. This value should always be '0'.
15:08	Message Size (MSIZE): Specifies the size of the message in bytes. The message size does not include the one Dword header. A value of '0' is invalid. The message size is always 4 bytes.
07:00	Reserved

The SAF-TE, SES-2, and SGPIO message formats are defined in the corresponding specifications. The LED message type is defined in Section 4.15.9.3. It is the responsibility of software to ensure the content of the message format is correct. If the message type is not programmed as 'LED' for this controller, the controller shall not take any action to update its LEDs. For LED message type, the message size is always four bytes.

4.15.9.3 LED Message Type

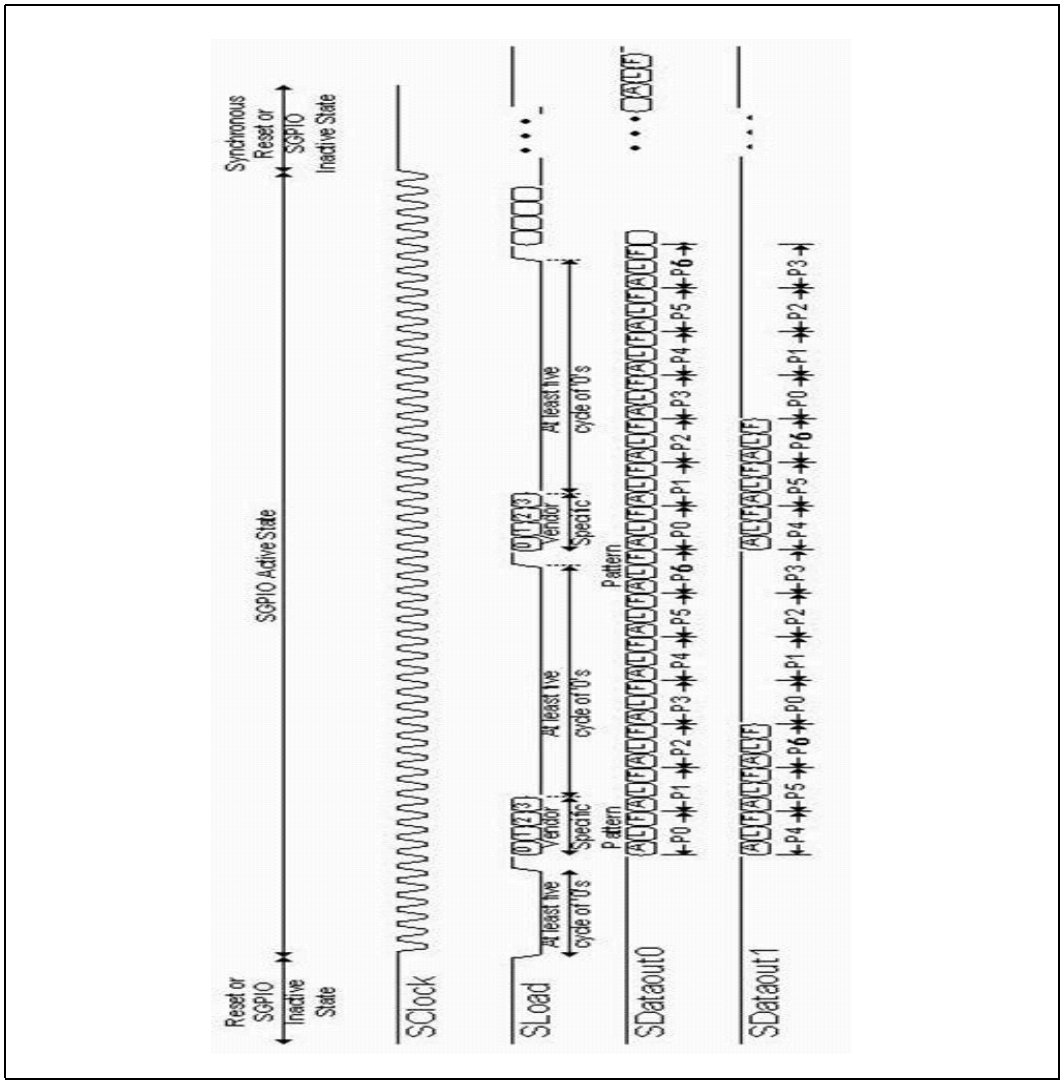
The LED message type specifies the status of up to three LEDs. Typically, the usage for these LEDs is activity, fault, and locate. Not all implementations necessarily contain all LEDs (for example, some implementations may not have a locate LED). The message identifies the HBA port number that the slot status applies to. The format of the LED message type is defined in Table 4-54. The LEDs shall retain their values until there is a following update for that particular slot.

Table 4-54. Multi-Activity LED Message Type

Byte	Description
3-2	Value (VAL) - This field describes the state of each LED for a particular location. There are three LEDs that may be supported by the HBA. Each LED has 3 bits of control. LED values are: 000b - LED shall be off 001b - LED shall be solid on as perceived by human eye All other values reserved The LED bit locations are: Bits 2:0 - Activity LED (may be driven by hardware) Bits 5:3 - Vendor Specific LED (for example, locate) Bits 8:6 - Vendor Specific LED (for example, fault) Bits 15:9 - Reserved Vendor specific message is: Bit 3:0 - Vendor Specific Pattern Bit 15:4 - Reserved ***Note:*** If Activity LED Hardware Driven (ATTR.ALHD) bit is set, host outputs the hardware LED value sampled internally and ignores software written activity value on bit [2:0]. Since the Enclosure Management does not support port multiplier based LED message, the LED message is generated independently based on respective port's operation activity. Vendor specific LED values Locate (Bits 5:3) and Fault (Bits 8:6) always are driven by software.
1	Port Multiplier Information - Port Multiplier not supported.
0	HBA Information - Specifies slot specific information related to the HBA. Note: This feature is Not Supported. Bits 4:0 - HBA port number for the slot that requires the status update. Bit 5 - If set to '1', Value is a vendor specific message that applies to the entire enclosure. If cleared to '0', Value applies to the port specified in bits 4:0. Bits 7:6 - Reserved

4.15.9.4 SGPIO Waveform

Figure 4-21. Serial Data Transmitted Over the SGPIO Interface



4.16 High Precision Event Timers

This function provides a set of timers that can be used by the operating system. The timers are defined such that in the future, the operating system may be able to assign specific timers to be used directly by specific applications. Each timer can be configured to cause a separate interrupt.

The system provides eight timers. The timers are implemented as a single counter each with its own comparator and value register. This counter increases monotonically. Each individual timer can generate an interrupt when the value in its value register matches the value in the main counter.

The registers associated with these timers are mapped to a memory space (much like the I/O APIC). However, it is not implemented as a standard PCI function. The BIOS reports to the operating system the location of the register space. The hardware can

support an assignable decode space; however, the BIOS sets this space prior to handing it over to the operating system. It is not expected that the operating system moves the location of these timers once it is set by the BIOS.

4.16.1 Timer Accuracy

1. The timers are accurate over any 1 ms period to within 0.05% of the time specified in the timer resolution fields.
2. Within any 100 microsecond period, the timer reports a time that is up to two ticks too early or too late. Each tick is less than or equal to 100 ns, so this represents an error of less than 0.2%.
3. The timer is monotonic. It does not return the same value on two consecutive reads (unless the counter has rolled over and reached the same value).

The main counter is clocked by the 14.31818 MHz clock, synchronized into the 66.666 MHz domain. This results in a non-uniform duty cycle on the synchronized clock, but does have the correct average period. The accuracy of the main counter is as accurate as the 14.31818 MHz clock.

4.16.2 Interrupt Mapping

Mapping Option #1 (Legacy Replacement Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is set. This forces the mapping found in Table 4-55.

Table 4-55. Legacy Replacement Routing

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	In this case, the 8254 timer does not cause any interrupts
1	IRQ8	IRQ8	In this case, the RTC does not cause any interrupts.
2 & 3	Per IRQ Routing Field.	Per IRQ Routing Field	
4, 5, 6, 7	not available	not available	

Mapping Option #2 (Standard Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is 0. Each timer has its own routing control. The interrupts can be routed to various interrupts in the 8259 or I/O APIC. A capabilities field indicates which interrupts are valid options for routing. If a timer is set for edge-triggered mode, the timers should not be share with any PCI interrupts.

The only supported interrupt values are as follows:

- Timer 0 and 1: IRQ20, 21, 22 & 23 (I/O APIC only).
- Timer 2: IRQ11 (8259 or I/O APIC) and IRQ20, 21, 22 & 23 (I/O APIC only).
- Timer 3: IRQ12 (8259 or I/O APIC) and IRQ 20, 21, 22 & 23 (I/O APIC only).
- Interrupts from Timer 4, 5, 6, 7 can only be delivered via direct FSB interrupt messages.

4.16.3 Periodic Versus Non-Periodic Modes

Non-Periodic Mode

Timer 0 is configurable to 32 (default) or 64-bit mode, whereas Timers 1, 2, and 3 only support 32-bit mode.

All of the timers support non-periodic mode.

See Section 2.3.9.2.1 of the IA-PC HPET Specification for a description of this mode.

Periodic Mode

Timer 0 is the only timer that supports periodic mode. See Section 2.3.9.2.2 of the *IA-PC HPET Specification* for a description of this mode.

The following usage model is expected:

1. Software clears the ENABLE_CNF bit to prevent any interrupts.
2. Software clears the main counter by writing a value of 00h to it.
3. Software sets the TIMER0_VAL_SET_CNF bit.
4. Software writes the new value in the TIMER0_COMPARATOR_VAL register
5. Software sets the ENABLE_CNF bit to enable interrupts.

The Timer 0 Comparator Value register cannot be programmed reliably by a single 64-bit write in a 32-bit environment except if only the periodic rate is being changed during run-time. If the actual Timer 0 Comparator Value needs to be reinitialized, then the following software solution always works regardless of the environment:

1. Set TIMER0_VAL_SET_CNF bit.
2. Set the lower 32 bits of the Timer0 Comparator Value register.
3. Set TIMER0_VAL_SET_CNF bit.
4. Set the upper 32 bits of the Timer0 Comparator Value register.

4.16.4 Enabling the Timers

The BIOS or operating system PnP code should route the interrupts. This includes the Legacy Rout bit, Interrupt Rout bit (for each timer), interrupt type (to select the edge or level type for each timer)

The Device Driver code should do the following for an available timer:

1. Set the Overall Enable bit (Offset 10h, bit 0).
2. Set the timer type field (selects one-shot or periodic).
3. Set the interrupt enable.
4. Set the comparator value.

4.16.5 Interrupt Levels

Interrupts directed to the internal 8259s are active high. See Section 4.7 for information regarding the polarity programming of the I/O APIC for detecting internal interrupts.

If the interrupts are mapped to the 8259 or I/O APIC and set for level-triggered mode, they can be shared with PCI interrupts. They may be shared although it’s unlikely for the operating system to attempt to do this.

If more than one timer is configured to share the same IRQ (using the TIMERn_INT_ROUT_CNF fields), then the software must configure the timers to level-triggered mode. Edge-triggered interrupts cannot be shared.

4.16.6 Handling Interrupts

If each timer has a unique interrupt and the timer has been configured for edge-triggered mode, then there are no specific steps required. No read is required to process the interrupt.

If a timer has been configured to level-triggered mode, then its interrupt must be cleared by the software. This is done by reading the interrupt status register and writing a 1 back to the bit position for the interrupt to be cleared.

Independent of the mode, software can read the value in the main counter to see how time has passed between when the interrupt was generated and when it was first serviced. If Timer 0 is set up to generate a periodic interrupt, the software can check to see how much time remains until the next interrupt by checking the timer value register.

4.16.7 Issues Related to 64-Bit Timers with 32-Bit Processors

A 32-bit timer can be read directly using processors that are capable of 32-bit or 64-bit instructions. However, a 32-bit processor may not be able to directly read 64-bit timer. A race condition comes up if a 32-bit processor reads the 64-bit register using two separate 32-bit reads. The danger is that just after reading one half, the other half rolls over and changes the first half.

If a 32-bit processor needs to access a 64-bit timer, it must first halt the timer before reading both the upper and lower 32-bits of the timer. If a 32-bit processor does not want to halt the timer, it can use the 64-bit timer as a 32-bit timer by setting the TIMERn_32MODE_CNF bit. This causes the timer to behave as a 32-bit timer. The upper 32-bits are always 0.

Alternatively, software may do a multiple read of the counter while it is running. Software can read the high 32 bits, then the low 32 bits, the high 32 bits again. If the high 32 bits have not changed between the two reads, then a rollover has not happened and the low 32 bits are valid. If the high 32 bits have changed between reads, then the multiple reads are repeated until a valid read is performed.

On a 64-bit platform, if software attempts a 64 bit read of the 64-bit counter, software must be aware that some platforms may split the 64 bit read into two 32 bit reads. The read maybe inaccurate if the low 32 bits roll over between the high and low reads.

4.17 USB* EHCI Host Controllers (B0:D29:F0)

The system contains one Enhanced Host Controller Interface (EHCI) host controller which support up to six USB 2.0 high-speed root ports. USB* 2.0 allows data transfers up to 480 Mb/s. USB* 2.0 based Debug Port is also implemented.

The following table summarizes the key features of the EHCI host controller.

Table 4-56. USB* EHCI Features

Parameter	USB* EHCI
Accessible by	Memory Space
Memory Data Structure	Separated into Periodic and Asynchronous lists
Differential Signaling Voltage	400 mV
# of Ports	6

4.17.1 EHC Initialization

Section 4.17.1.1, "BIOS Initialization" through Section 4.17.1.3, "EHC Resets" describe the expected Enhanced Host Controller (EHC) initialization sequence in chronological order, beginning with a complete power cycle in which the suspend well and core well have been off.

4.17.1.1 BIOS Initialization

BIOS performs a number of platform customization steps after the core well has powered up.

Note: Contact your Intel Field Representative for additional BIOS information.

4.17.1.2 Driver Initialization

See Chapter 4 of the *Enhanced Host Controller Interface Specification for Universal Serial Bus,* Revision 1.0.

4.17.1.3 EHC Resets

In addition to the standard hardware resets, portions of the EHC are reset by the HCRESET bit and the transition from the $D3_{HOT}$ device power management state to the D0 state. The effects of each of these resets are:

Reset	Does Reset	Does not Reset	Comments
HCRESET bit set.	Memory space registers except Structural Parameters (which is written by BIOS).	Configuration registers.	The HCRESET must only affect registers that the EHCI driver controls. PCI Configuration space and BIOS-programmed parameters can not be reset.
Software writes the Device Power State from $D3_{HOT}$ (11b) to D0 (00b).	Core well registers (except BIOS-programmed registers).	Suspend well registers; BIOS-programmed core well registers.	The D3-to-D0 transition must not cause wake information (suspend well) to be lost. It also must not clear BIOS-programmed registers because BIOS may not be invoked following the D3-to-D0 transition.

If the detailed register descriptions give exceptions to these rules, those exceptions override these rules. This summary is provided to explain the reasons for the reset policies.

4.17.2 Data Structures in Main Memory

See Section 3 and Appendix B of the *Enhanced Host Controller Interface Specification for Universal Serial Bus,* Revision 1.0 for details.

4.17.3 USB* 2.0 Enhanced Host Controller DMA

The USB* 2.0 EHC implements three sources of USB* packets. They are, in order of priority on USB* during each microframe:

1. The USB* 2.0 Debug Port
2. The Periodic DMA engine
3. The Asynchronous DMA engine

The system always performs any currently-pending debug port transaction at the beginning of a microframe, followed by any pending periodic traffic for the current microframe. If there is time left in the microframe, then the EHC performs any pending asynchronous traffic until the end of the microframe (EOF1). The debug port traffic is only presented on Port #1. The other ports are idle during this time.

4.17.4 Data Encoding and Bit Stuffing

See Chapter 8 of the *Universal Serial Bus Specification, Revision 2.0.*

4.17.5 Packet Formats

See Chapter 8 of the *Universal Serial Bus Specification, Revision 2.0*. The EHCI allows entrance to USB* test modes, as defined in the USB* 2.0 specification, including Test J, Test Packet, etc. However, Test Packet test mode interpacket gap timing may not meet the USB* 2.0 specification.

4.17.6 USB* 2.0 Interrupts and Error Conditions

Section 4 of the *Enhanced Host Controller Interface Specification for Universal Serial Bus,* Revision 1.0 goes into detail on the EHC interrupts and the error conditions that cause them. All error conditions that the EHC detects can be reported through the EHCI Interrupt status bits. Only specific interrupt and error-reporting behavior is documented in this section. The EHCI Interrupts section must be read first, followed by this section of the datasheet to fully comprehend the EHC interrupt and error-reporting functionality.

- Based on the EHC's buffer sizes and buffer management policies, the Data Buffer Error may never occur.
- Master Abort and Target Abort responses from hub interface on EHC-initiated read packets is treated as Fatal Host Errors. The EHC halts when these conditions are encountered.
- The system may assert the interrupts which are based on the interrupt threshold as soon as the status for the last complete transaction in the interrupt interval has been posted in the internal write buffers. The requirement in the *Enhanced Host Controller Interface Specification for Universal Serial Bus,* Revision 1.0 (that the status is written to memory) is met internally, even though the write may not be seen on DMI before the interrupt is asserted.
- Since the system supports the 1024-element Frame List size, the Frame List Rollover interrupt occurs every 1024 milliseconds.
- The system delivers interrupts using PIRQH#.
- The system does not modify the CERR count on an Interrupt IN when the "Do Complete-Split" execution criteria are not met.
- For complete-split transactions in the Periodic list, the "Missed Microframe" bit does not get set on a control-structure-fetch that fails the late-start test. If subsequent accesses to that control structure do not fail the late-start test, then the "Missed Microframe" bit is set and written back.

4.17.6.1 Aborts on USB* 2.0-Initiated Memory Reads

If a read initiated by the EHC is aborted, the EHC treats it as a fatal host error. The following actions are taken when this occurs:

- The Host System Error status bit is set
- The DMA engines are halted after completing up to one more transaction on the USB* interface
- If enabled (by the Host System Error Enable), then an interrupt is generated
- If the status is Master Abort, then the Received Master Abort bit in configuration space is set
- If the status is Target Abort, then the Received Target Abort bit in configuration space is set
- If enabled (by the SERR Enable bit in the function's configuration space), then the Signaled System Error bit in configuration bit is set.

4.17.7 USB* 2.0 Power Management

4.17.7.1 Pause Feature

This feature allows platforms to dynamically enter low-power states during brief periods when the system is idle (for example, between keystrokes). This is useful for enabling power management features. The policies for entering these states typically are based on the recent history of system bus activity to incrementally enter deeper power management states. Normally, when the EHC is enabled, it regularly accesses main memory while traversing the DMA schedules looking for work to do; this activity is viewed by the power management software as a non-idle system, thus preventing the power managed states to be entered. Suspending all of the enabled ports can prevent the memory accesses from occurring, but there is an inherent latency overhead with entering and exiting the suspended state on the USB* ports that makes this unacceptable for the purpose of dynamic power management. As a result, the EHCI software drivers are allowed to pause the EHC's DMA engines when it knows that the traffic patterns of the attached devices can afford the delay. The pause only prevents the EHC from generating memory accesses; the SOF packets continue to be generated on the USB* ports (unlike the suspended state).

4.17.7.2 Suspend Feature

The *Enhanced Host Controller Interface (EHCI) For Universal Serial Bus Specification*, Section 4.3 describes the details of Port Suspend and Resume.

4.17.7.3 ACPI Device States

The USB* 2.0 function only supports the D0 and D3 PCI Power Management states.

Notes regarding the implementation of the Device States:

- The EHC hardware does not inherently consume any more power when it is in the D0 state than it does in the D3 state. However, software is required to suspend or disable all ports prior to entering the D3 state such that the maximum power consumption is reduced.
- In the D0 state, all implemented EHC features are enabled.
- In the D3 state, accesses to the EHC memory-mapped I/O range performs a master abort. Since the Debug Port uses the same memory range, the Debug Port is only operational when the EHC is in the D0 state.

- In the D3 state, the EHC interrupt must never assert for any reason. The internal PME# signal is used to signal wake events, etc.
- When the Device Power State field is written to D0 from D3, an internal reset is generated. See section EHC Resets for general rules on the effects of this reset.
- Attempts to write any other value into the Device Power State field other than 00b (D0 state) and 11b (D3 state) completes normally without changing the current value in this field.

4.17.7.4 ACPI System States

The EHC behavior as it relates to other power management states in the system is summarized in the following list:

- The system is always in the S0 state when the EHC is in the D0 state. However, when the EHC is in the D3 state, the system may be in any power management state (including S0).
- When in D0, the Pause feature (See Section 4.17.7.1) enables dynamic processor low-power states to be entered.
- The PLL in the EHC is disabled when entering the S3/S4/S5 states (core power turns off).
- All core well logic is reset in the S3/S4/S5 states.

4.17.8 USB* 2.0 Legacy Keyboard Operation

The system must support the possibility of a keyboard downstream from either a full-speed/low-speed or a high-speed port. The description of the legacy keyboard support is unchanged from USB* 1.1.

The EHC provides the basic ability to generate SMIs on an interrupt event, along with more sophisticated control of the generation of SMIs.

4.17.9 USB* 2.0 Based Debug Port

The system supports the elimination of the legacy COM ports by providing the ability for new debugger software to interact with devices on a USB* 2.0 port.

High-level restrictions and features are:

- Operational before USB* 2.0 drivers are loaded.
- Functions even when the port is disabled.
- Allows normal system USB* 2.0 traffic in a system that may only have one USB* port.
- Debug Port device (DPD) must be high-speed capable and connect directly to Port #1 of the system (for example, the DPD cannot be connected to Port #1 through a hub. When a DPD is detected, the EHCI bypasses the integrated Rate Matching Hub and connect directly to the port and the DPD.).
- Debug Port FIFO always makes forward progress (a bad status on USB* is simply presented back to software).
- The Debug Port FIFO is only given one USB* access per microframe.

The Debug port facilitates operating system and device driver debug. It allows the software to communicate with an external console using a USB* 2.0 connection. Because the interface to this link does not go through the normal USB* 2.0 stack, it allows communication with the external console during cases where the operating system is not loaded, the USB* 2.0 software is broken, or where the USB* 2.0 software is being debugged. Specific features of this implementation of a debug port are:

- Only works with an external USB* 2.0 debug device (console)
- Implemented for a specific port on the host controller
- Operational anytime the port is not suspended AND the host controller is in D0 power state.
- Capability is interrupted when port is driving USB* RESET

4.17.9.1 Theory of Operation

There are two operational modes for the USB* debug port:

1. Mode 1 is when the USB* port is in a disabled state from the viewpoint of a standard host controller driver. In Mode 1, the Debug Port controller is required to generate a "keepalive" packets less than 2 ms apart to keep the attached debug device from suspending. The keepalive packet should be a standalone 32-bit SYNC field.
2. Mode 2 is when the host controller is running (for example, host controller's *Run/Stop#* bit is 1). In Mode 2, the normal transmission of SOF packets keeps the debug device from suspending.

Behavioral Rules

1. In both Modes 1 and 2, the Debug Port controller must check for software requested debug transactions at least every 125 microseconds.
2. If the debug port is enabled by the debug driver, and the standard host controller driver resets the USB* port, USB* debug transactions are held off for the duration of the reset and until after the first SOF is sent.
3. If the standard host controller driver suspends the USB* port, then USB* debug transactions are held off for the duration of the suspend/resume sequence and until after the first SOF is sent.
4. The ENABLED_CNT bit in the debug register space is independent of the similar port control bit in the associated Port Status and Control register.

Table 4-57 shows the debug port behavior related to the state of bits in the debug registers as well as bits in the associated Port Status and Control register.

Table 4-57. Debug Port Behavior (Sheet 1 of 2)

OWNER_CNT	ENABLED_CT	Port Enable	Run / Stop	Suspend	Debug Port Behavior
0	X	X	X	X	Debug port is not being used. Normal operation.
1	0	X	X	X	Debug port is not being used. Normal operation.
1	1	0	0	X	Debug port in Mode 1. SYNC keepalives sent plus debug traffic
1	1	0	1	X	Debug port in Mode 2. SOF (and only SOF) is sent as keepalive. Debug traffic is also sent. No other normal traffic is sent out this port because the port is not enabled.
1	1	1	0	0	Illegal. Host controller driver should never put controller into this state (enabled, not running and not suspended).

Table 4-57. Debug Port Behavior (Sheet 2 of 2)

OWNER_CNT	ENABLED_CT	Port Enable	Run / Stop	Suspend	Debug Port Behavior
1	1	1	0	1	Port is suspended. No debug traffic sent.
1	1	1	1	0	Debug port in Mode 2. Debug traffic is interspersed with normal traffic.
1	1	1	1	1	Port is suspended. No debug traffic sent.

4.17.9.1.1 OUT Transactions

An Out transaction sends data to the debug device. It can occur only when the following are true:

- The debug port is enabled
- The debug software sets the GO_CNT bit
- The WRITE_READ#_CNT bit is set

The sequence of the transaction is as follows:

1. Software sets the appropriate values in the following bits:
   - USB_ADDRESS_CNF
   - USB_EndPoint_CNF
   - DATA_BUFFER[63:0]
   - TOKEN_PID_CNT[7:0]
   - SEND_PID_CNT[15:8]
   - DATA_LEN_CNT
   - WRITE_READ#_CNT: (This is always 1 for OUT transactions)
   - GO_CNT: (This is always 1 to initiate the transaction)
2. The debug port controller sends a token packet consisting of the following content:
   - SYNC
   - TOKEN_PID_CNT field
   - USB_ADDRESS_CNT field
   - USB_EndPoint_CNT field
   - 5-bit CRC field
3. After sending the token packet, the debug port controller sends a data packet consisting of the following content:
   - SYNC
   - SEND_PID_CNT field
   - The number of data bytes indicated in DATA_LEN_CNT from the DATA_BUFFER
   - 16-bit CRC

Note: A DATA_LEN_CNT value of 0 is valid in which case no data bytes would be included in the packet.

4. After sending the data packet, the controller waits for a handshake response from the debug device.

- If a handshake is received, the debug port controller:

a. Places the received PID in the RECEIVED_PID_STS field
b. Resets the ERROR_GOOD#_STS bit
c. Sets the DONE_STS bit

- If no handshake PID is received, the debug port controller:
  a. Sets the EXCEPTION_STS field to 001b
  b. Sets the ERROR_GOOD#_STS bit
  c. Sets the DONE_STS bit

4.17.9.1.2 IN Transactions

An IN transaction receives data from the debug device. An IN transaction can occur only when the following states are true:

- The debug port is enabled
- The debug software sets the GO_CNT bit
- The WRITE_READ#_CNT bit is reset

The sequence of the transaction takes place as follows:

1. Software sets the appropriate values in the following bits:
   — USB_ADDRESS_CNF
   — USB_EndPoint_CNF
   — TOKEN_PID_CNT[7:0]
   — DATA_LEN_CNT
   — WRITE_READ#_CNT: (This is always 0 for IN transactions)
   — GO_CNT: (This always 1 to initiate the transaction)
2. The debug port controller sends a token packet consisting of the following content:
   — SYNC
   — TOKEN_PID_CNT field
   — USB_ADDRESS_CNT field
   — USB_EndPoint_CNT field
   — 5-bit CRC field
3. After sending the token packet, the debug port controller waits for a response from the debug device. If a response is received:
   — The received PID is placed into the RECEIVED_PID_STS field
   — Any subsequent bytes are placed into the DATA_BUFFER
   — The DATA_LEN_CNT field is updated to show the number of bytes that were received after the PID
4. If a valid packet was received from the device that was one byte in length (indicating it was a handshake packet), then the debug port controller performs the following actions:
   — Resets the ERROR_GOOD#_STS bit
   — Sets the DONE_STS bit
5. If a valid packet was received from the device that was more than one byte in length (indicating it was a data packet), then the debug port controller performs the following actions:
   — Transmits an ACK handshake packet

— Resets the ERROR_GOOD#_STS bit

— Sets the DONE_STS bit

6. If no valid packet is received, then the debug port controller performs the following actions:

— Sets the EXCEPTION_STS field to 001b

— Sets the ERROR_GOOD#_STS bit

— Sets the DONE_STS bit

4.17.9.1.3 Debug Software

Enabling the Debug Port

Debug software must address two mutually exclusive conditions as part of its startup processing:

- The EHCI has been initialized by system software
- The EHCI has not been initialized by system software

Debug software can determine the current 'initialized' state of the EHCI by examining the Configure Flag in the EHCI USB* 2.0 Command Register. If this flag is set, then system software has initialized the EHCI. Otherwise the EHCI should not be considered initialized. Debug software initializes the debug port registers depending on the state of the EHCI. Before this can be accomplished, debug software must determine which root USB* port is designated as the debug port.

Determining the Debug Port

Debug software can determine which USB* root port has been designated as the debug port by examining bits 20:23 of the EHCI Host Controller Structural Parameters register. This 4-bit field represents the numeric value assigned to the debug port (for example, 0001=port 1).

Debug Software Startup with Non-Initialized EHCI

Debug software can attempt to use the debug port if after setting the OWNER_CNT bit, the Current Connect Status bit in the appropriate (See Determining the Debug Port) PORTSC register is set. If the Current Connect Status bit is not set, then debug software may choose to terminate or it may choose to wait until a device is connected.

If a device is connected to the port, then debug software must reset/enable the port. Debug software does this by setting and then clearing the Port Reset bit the PORTSC register. To guarantee a successful reset, debug software should wait at least 50 ms before clearing the Port Reset bit. Due to possible delays, this bit may not change to 0 immediately; reset is complete when this bit reads as 0. Software must not continue until this bit reads 0.

If a high-speed device is attached, then the EHCI automatically sets the Port Enabled/Disabled bit in the PORTSC register and the debug software can proceed. Debug software should set the ENABLED_CNT bit in the Debug Port Control/Status register, and then reset (clear) the Port Enabled/Disabled bit in the PORTSC register so the system host controller driver does not see an enabled port when it is first loaded.

Debug Software Startup with Initialized EHCI

Debug software can attempt to use the debug port if the Current Connect Status bit in the appropriate (See Determining the Debug Port) PORTSC register is set. If the Current Connect Status bit is not set, then debug software may choose to terminate or it may choose to wait until a device is connected. If a device is connected, then debug software must set the OWNER_CNT bit and then the ENABLED_CNT bit in the Debug Port Control/Status register.

Determining Debug Peripheral Presence

After enabling the debug port functionality, debug software can determine if a debug peripheral is attached by attempting to send data to the debug peripheral. If all attempts result in an error (Exception bits in the Debug Port Control/Status register indicates a Transaction Error), then the attached device is not a debug peripheral. If the debug port peripheral is not present, then debug software may choose to terminate or it may choose to wait until a debug peripheral is connected.

4.17.10 EHCI Caching

EHCI Caching is a power management feature in the USB* (EHCI) host controllers which enables the controller to execute the schedules entirely in cache and eliminates the need for the DMA engine to access memory when the schedule is idle. EHCI caching allows the processor to maintain longer C-state residency times and provides substantial system power savings.

4.17.11 USB* Pre-Fetch Based Pause

The Pre-Fetch Based Pause is a power management feature in USB* (EHCI) host controllers to ensure maximum C3/C4 processor power state time with C2 popup. This feature applies to the period schedule, and works by allowing the DMA engine to identify periods of idleness and preventing the DMA engine from accessing memory when the periodic schedule is idle. Typically in the presence of periodic devices with multiple millisecond poll periods, the periodic schedule is idle for several frames between polls.

The USB* Pre-Fetch Based Pause feature is disabled by setting bit 4 of EHCI Configuration Register.

4.17.12 USB* Overcurrent Protection

The system has implemented programmable USB* Overcurrent signals. There are a total of four overcurrent pins shared across the six ports.

Four overcurrent signals have been allocated to the ports in each USB* device:

- OC[3:0]# for Device 29 (Ports 0-5)

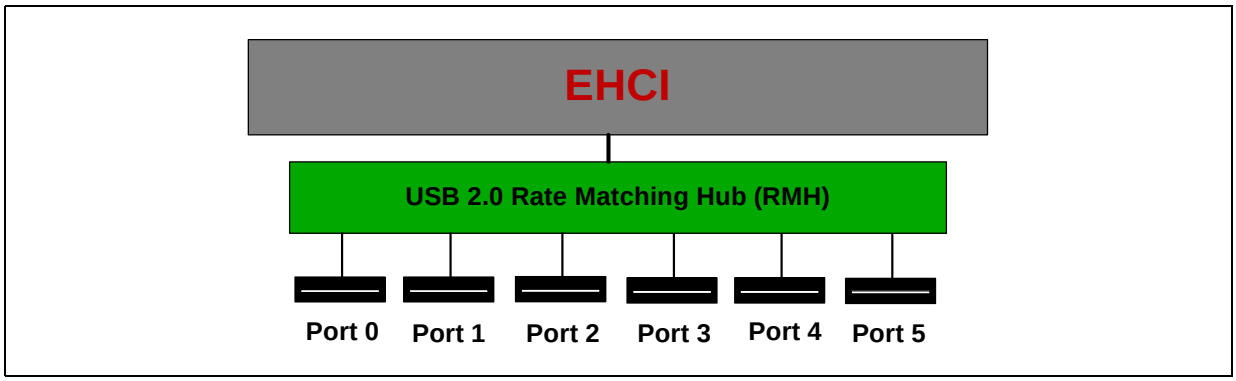
Each pin is mapped to one or more ports by setting bits in the USB*OCM1 register. See Section 7.1.1.68. It is system BIOS' responsibility to ensure that each port is mapped to only one over current pin. Operation with more than one overcurrent pin mapped to a port is undefined. It is expected that multiple ports are mapped to a single overcurrent pin, however they should be connected at the port and not at the pin. Shorting these pins together may lead to reduced test capabilities.

4.18 Integrated USB* 2.0 Rate Matching Hub

The system has an integrated USB* 2.0 Rate Matching Hub (RMH). The Hub is connected to the EHCI controller as shown in Figure 4-22. The Hub convert low and full-speed traffic into high-speed traffic. When the RMH is enabled, it appears to software like an external hub is connected to Port 0 of each EHCI controller. In addition, port 1 the RMH is muxed with Port 1 of the EHCI controller and is able to bypass the RMH for use as the Debug Port.

The hub operates like any USB* 2.0 Discrete Hub and consumes one tier of hubs allowed by the section 4.1.1 in the USB* 2.0 specification. A maximum of four additional non-root hubs can be supported on any of the USB* Ports. The RMH reports Vendor ID = 0x8087 and Product ID = 0x0020.

Figure 4-22. EHCI with USB* 2.0 with Rate Matching Hub



4.18.1 Architecture

A hub consists of three components: the Hub Repeater, the Hub Controller, and the Transaction Translator.

- The Hub Repeater is responsible for connectivity setup and tear-down. It also supports exception handling, such as bus fault detection and recovery and connect/disconnect detect.
- The Hub Controller provides the mechanism for host-to-hub communication. Hub-specific status and control commands permit the host to configure a hub and to monitor and control its individual downstream facing ports.
- The Transaction Translator (TT) responds to high-speed split transactions and translates them to full-/low-speed transactions with full-/low-speed devices attached on downstream facing ports. There is 1 TT per RMH.

See Chapter 11 of the USB* 2.0 Specification for more details on the architecture of the hubs.

4.19 Thermal Management

4.19.1 Modes of Operation

The PCH thermal sensor is available in Non-EndPoint and Normal modes. It has two modes of operation:

- the analog mode (for example, sequencer disabled)

- the four-function mode, which includes a digital thermometer.

The analog mode is kept primarily as a backup in case thermal time constants are found to be significantly faster than the temperature trip point detection rate of the newer four-function mode or the four-function mode has functional problems. Also, it is possible that the four-function mode has no utility beyond that of catastrophic/hot/aux detection, depending on the thermal sensor accuracy requirements of other usage models.

The analog mode uses two comparators and only reports catastrophic and/or hot trips.

The four-function mode uses one comparator that is time-multiplexed to perform 4 functions. A state machine switches the specific function of the thermal sensor and logic at a rate dependent on the Thermal Sensor Control [Sequencer Enable and Rate] register setting, giving a separate output reading for each function. The sequencer can be configured in different modes to support various validation scenarios and fall-back modes. See the Thermal Sensor Control register (TSC) for detailed descriptions of the various modes and associated thermal sensor settling times.

The five functions of the "four-function mode" are:

- Catastrophic Trip Point - This trip point is set at the temperature at which the chip must be shut down immediately without any software support. For this logic to function, the catastrophic trip point must correspond to a temperature guaranteed to be functional. Special care must be taken to make sure that the sequencer logic, Power Management logic, and THRMTRIP# are functional at the catastrophic trip point.
- Hot Temperature Trip Point - This trip point may be set dynamically if desired and provides an interrupt when it is crossed in either direction. Software could optionally set this as an "Interrupt me when the temperature goes above this level" setting.
- Auxiliary Temperature Trip Point - This trip point is set below the Hot Temperature Trip Point and provides the same response options. However, the responses are separately programmable from the Hot Temp in order to provide incrementally more aggressive actions. The Aux trip point is fully software-programmable during system run time.
- Auxiliary2 (Aux2) Temperature Trip Point - This trip point is typically set below the Aux Temperature Trip Point and provides interrupt generation capability. The Aux2 trip point is fully software-programmable during system run time.
- Thermometer - The thermometer is implemented via a counter that starts at 0 and increments during each sample point until the comparator indicates that the temperature is above the current value. The value of the counter is loaded into a read-only register when the comparator first trips.

4.19.2 Thermal Reporting Over System Management Link 1 Interface (SMLink1)

SMLink1 is the SMBus interface utilized by an External Management Controller (BMC/EC) for Platform Thermal Reporting. It is used by the BMC/EC to control or obtain thermal sensor data from DIMMs, CPU, PCH, and other components integrated in the system.

The Thermal Reporting features, functions, and interface configurations are described in the *SMBus-to-PECI Bridge Protocol Application Note (Doc# 460415)*.

4.20 WatchDog Timer (WDT) Host Controller (B0:D31:F7)

The Watchdog Host Controller device implements per-thread reset capability. The controller manifests itself as a PCI Express* legacy EndPoint. This device is MSI-X compatible.

4.20.1 Theory Of Operation

The reset WDTs provide a mechanism for threads to be woken from a hung condition. A WDT is provisioned for each thread on the platform. Some platforms does not use/enable all the WDTs. Each WDT can be loaded with a specific value and allowed to decrement. The time taken to fully decrement is referred to as the "interval" hereafter.

The WDTnCOUNT registers stores the 32 bit reload value for the counter. The lower 10 bits of the reload value will always be zero.

4.20.2 Watchdog Timer Behavior

Each WDT behaves in an identical fashion. Each is clocked off a 10 bit prescaler counter. The prescaler counter is free running and cannot be disabled. The prescaler will run off the 125MHz backbone clock allowing a prescaler rollover time of up to 1K * 8ns = 8us. If the prescaler value is set to zero the individual WDTs will decrement at the 125MHz rate, otherwise the WDTs will decrement when the prescaler rolls over.

Each 32 bit WDT counter can extend the interval to several hours (8e-06 * 2^{32} ~ 32 Ksec). Expected usage is in the ms/second range. Given that the minimum valid value of a reload register is 0x400, the minimum interval that can be generated is ~8 us.

When enabled properly and reloaded the WDT will start decrementing (under control of the prescaler) immediately and will continue to do so until it reaches zero unless it is reloaded (by a write to WDTnCMD) during that time. A reload will restart the WDT immediately if enabled.

There are two events (and corresponding interrupts) that the unit generates when counting down:

1. The WARNing event is generated at approximately the halfway stage (when the counter value = WDTnCOUNT[31:1]). This results in an interrupt that should be sufficient to cause most threads to act.
2. The RESET event is generated when the WDT counter transitions from 0x1 to 0x0. this is intended to be a more severe interrupt that possibly resets a hung thread.

4.20.3 Pending Bit Array

Each event will set a corresponding bit in the PBA. If the interrupt is not masked the device should attempt to deliver the corresponding interrupt as soon as possible after the event occurs. Interrupt delivery may of course be delayed by arbitration on the backbone bus.

If a RESET event occurs before the earlier WARN interrupt has been delivered, the WARN PBA bit is cleared and the WARN interrupt can be cancelled. The RESET bit is set and the RESET interrupt send.

When an interrupt is sent or cancelled, the PBA bit is cleared. Both the RESET and WARN PBA bits are cleared when the WDT is reloaded.

The interrupt mask does not affect the setting of the PBA bits.

4.20.4 MSI Interrupt Formation

The interrupts are generated as MSI interrupts. An MSI interrupt is a memory write PCIe* transaction. The device must deliver the interrupt as soon as possible after the corresponding PBA bit is set (unless the interrupt is masked by the Vector Control register or CMD.BME).

An MSI packet is a memory write txn. For a given RESET/WARN interrupt, the packet is built as a 3DW header (because the upper half of the address is 0x00000000) with data as shown in Figure 4-23.

Figure 4-23. MSI Packet Header

	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	R	10			0	0	0	0	0	R	TC=0			R			0	0	00		R				Length = 01							
4	Req ID = 0x00FF (Bus 0, Dev 31, Fn 7)																Tag = 0								0x0				0x3			
8	Lower Address – Contents of WDTnMAW/R																															

The single DW of data is the contents of the WDTnMDW/R register.

4.20.5 Usage Models

There are two models for setting up and initiating timer activity:

- BIOS sets timer values and all timers are enabled simultaneously:
  - BIOS configures the TBAR content & required WDTnCOUNT/WDTnCFG registers. Values can be locked if necessary.
  - Unused WDTs is left with WDTnCOUNT = 0x00, disabling them. These values can be locked if necessary.
  - BIOS sets up the prescaler WDT_PSCALE. Lock if necessary.
  - BIOS reloads (write to WDTnCMD) all operating timers – no counting because of global enable.
  - Later, OS globally enables all timers simultaneously using WDT_GBLCFG.GEN
  - Threads are thereafter expected to avoid interrupts by reloading the WDT.
- Threads set their timer values & kick off timers individually:
  - BIOS configures TBAR. Lock if necessary.
  - BIOS locks any unused WDTnCOUNT registers at 0x00 if necessary.
  - BIOS sets up the prescaler WDT_PSCALE. Lock if necessary.
  - BIOS globally enables using WDT_GBLCFG.GEN
  - Later on individual threads can set the WDTnCOUNT values (locking if necessary). The threads can then initiate timer activity by writing to WDTnCMD.
  - Threads are thereafter expected to avoid interrupts by reloading the WDT.

4.21 Serial Peripheral Interface (SPI) (B0:D31:F0)

The Serial Peripheral Interface (SPI) is a 4-pin interface. This 4-pin SPI interface consists of clock (CLK), master data out (Master Out Slave In (MOSI)), master data in (Master In Slave Out (MISO)) and an active low chip select (SPI_CS[1:0]#).

The system supports up to two SPI flash devices using two separate Chip Select pins. Each SPI flash device can be up to 16 MBytes. The SPI interface supports 20 MHz, 33 MHz and 50 MHz SPI devices. A SPI Flash device controlled with Chip Select 0 and with a valid descriptor MUST be attached directly to the system.

Communication on the SPI bus is done with a Master–Slave protocol. The Slave is connected to the system and is implemented as a tri-state bus.

SPI Flash has two operational modes, descriptor and non-descriptor. Non-Descriptor Mode is not supported. A valid flash descriptor is required for all platforms.

4.21.1 Descriptor Mode

Descriptor Mode is required for all platforms. It enables many system features:

- Intel® Management Engine Firmware
- PCI Express* root port configuration
- Supports up to two SPI components using two separate chip select pins
  - — Tow SPI Flash components or
  - — One SPI Flash and one user authentication device
- Hardware enforced security restricting master accesses to different regions
- Soft Strap regions provides the ability to use Flash Non-Volatile Memory (NVM)
- Supports the SPI Fast Read instruction and frequencies of up to 50 MHz
- Uses standardized Flash Instruction Set

4.21.2 SPI Flash Regions

In Descriptor Mode the Flash is divided into five separate regions:

Region	Content
0	Flash Descriptor
1	BIOS
2	Management Engine
3	Reserved
4	Platform Data

Only two masters can access the four accessible regions: Host processor running BIOS code and Management Engine. The only required region is Region 0, the Flash Descriptor. Region 0 must be located in the first sector of device 0 (offset 0).

Flash Region Sizes

SPI flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4 KB or larger block. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the ME and BIOS regions.

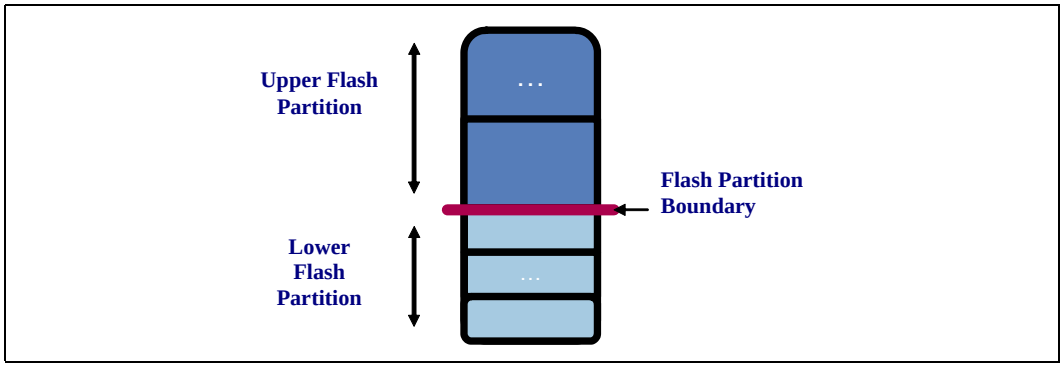
Table 4-58. Region Size Versus Erase Granularity of Flash Components

Region	Size with 4 KB Blocks	Size with 8 KB Blocks	Size with 64 KB Blocks
Descriptor	4 KB	8 KB	64 KB
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
ME	128 KB	128 KB	128 KB

4.21.2.1 Device Partitioning

The SPI Flash controller supports two sets of attributes in SPI flash space. This allows for supporting an asymmetric flash component that has two separate sets of attributes in the upper and lower part of the memory array. An example is a flash part that has different erase granularities in two different parts of the memory array. This allows for the usage of two separate flash vendors if using two different flash parts.

Figure 4-24. Flash Partition Boundary

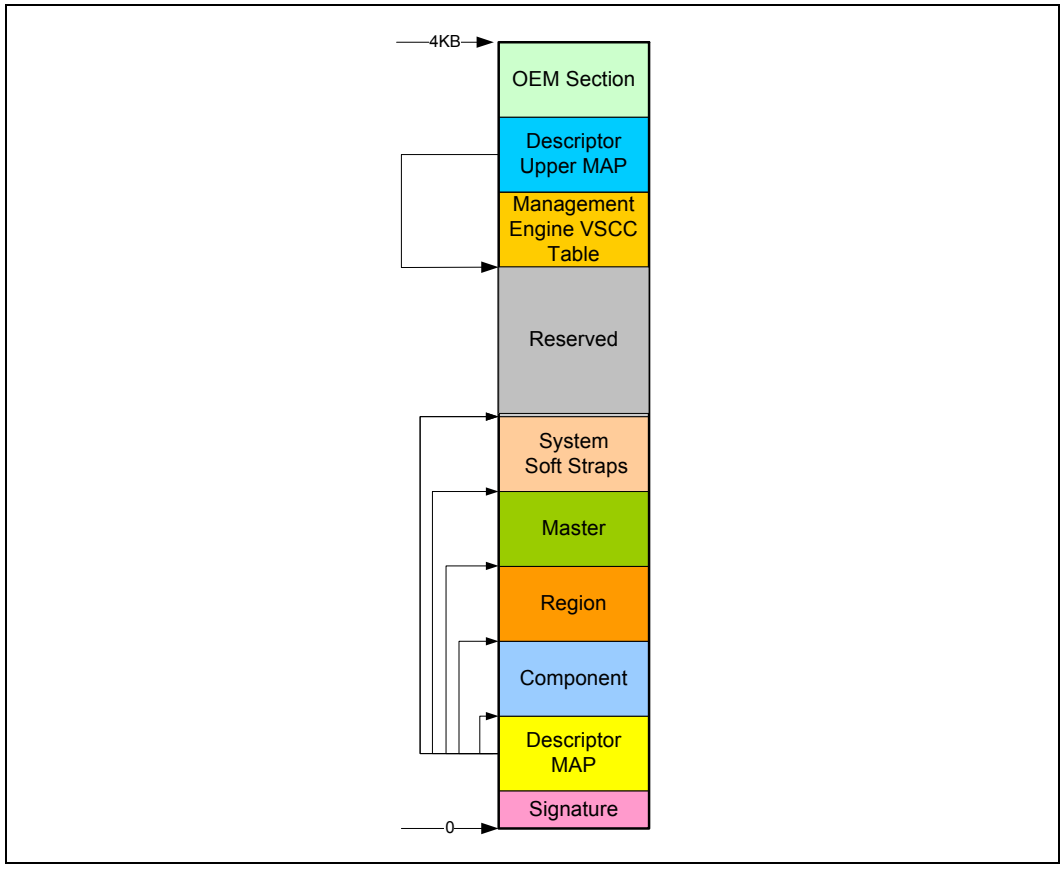


4.21.3 Flash Descriptor

The maximum size of the Flash Descriptor is 4 KB. If the block/sector size of the SPI flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. The flash descriptor requires its own block at the bottom of memory (0x00h). The information stored in the Flash Descriptor can only be written during the manufacturing process as its read/write permissions must be set to Read only when the system leaves the manufacturing floor.

The Flash Descriptor is made up of eleven sections:

Figure 4-25. Flash Descriptor Sections



1. The Flash Signature selects Descriptor Mode and verifies if the flash is programmed and functioning. The data at the bottom of the flash (offset 0) must be 0FF0A55Ah to be in Descriptor mode.
2. The Descriptor Map has pointers to the other five descriptor sections and the size of each.
3. The Component section has information about the SPI flash in the system including: the number of components, density of each, illegal instructions (such as chip erase), and frequencies for read, fast read and write/erase instructions.
4. The Region section points to the three other regions and the size of each region.
5. The Master Region contains the security settings for the flash, granting read/write permissions for each region and identifying each master by a requestor ID. See Section 4.21.3.1 for more information.
6. The Processor and System Soft Strap sections contain processor and system configurable parameters.
7. Same as #6.
8. The Reserved region between the top of the Processor Strap section and the bottom of the OEM Section is reserved for future chipset usages.
9. The Descriptor Upper MAP determines the length and base address of the Management Engine VSCC Table.

10. The Management Engine VSCC Table holds the JEDEC ID and the VSCC information of the entire SPI Flash supported by the NVM image.
11. OEM Section is 256 bytes reserved at the top of the Flash Descriptor for OEM use.

4.21.3.1 Descriptor Master Region

The master region defines read and write access setting for each region of the SPI device. The master region recognizes two masters: BIOS and Management Engine. Each master is only allowed to do direct reads of its primary regions.

Table 4-59. Region Access Control Table

Master Read/Write Access		
Region	**CPU and BIOS**	**ME**
Descriptor	N/A	N/A
BIOS	CPU and BIOS can always read from and write to BIOS Region	Read / Write
Management Engine	Read / Write	ME can always read from and write to ME Region
Platform Data Region	N/A	N/A

4.21.4 Flash Access

There are two types of flash accesses:

- Direct Access:
  - Masters are allowed to do direct read only of their primary region
  - Master's Host or Management Engine virtual read address is converted into the SPI Flash Linear Address (FLA) using the Flash Descriptor Region Base/Limit registers
- Program Register Access:
  - Program Register Accesses are not allowed to cross a 4KB boundary and can not issue a command that might extend across two components
  - Software programs the FLA corresponding to the region desired
  - Software must read the devices Primary Region Base/Limit address to create a FLA.

4.21.4.1 Direct Access Security

- Requester ID of the device must match that of the primary Requester ID in the Master Section
- Calculated Flash Linear Address must fall between primary region base/limit
- Direct Write not allowed
- Direct Read Cache contents are reset to 0's on a read from a different master
  - Supports the same cache flush mechanism in ICH7 which includes Program Register Writes

4.21.4.2 Register Access Security

- Only primary region masters can access the registers
- Masters are only allowed to read or write those regions they have read/write permission
- Using the Flash Region Access Permissions, one master can give another master read/write permissions to their area
- Using the five Protected Range registers, each master can add separate read/write protection above that granted in the Flash Descriptor for their own accesses
  - — Example: BIOS may want to protect different regions of BIOS from being erased
  - — Ranges can extend across region boundaries

4.21.5 Serial Flash Device Compatibility Requirements

A variety of serial flash devices exist in the market. For a serial flash device to be compatible with the SPI bus, it must meet the minimum requirements detailed in the following sections.

Note: All system platforms have the required Intel® Management engine firmware.

4.21.5.1 SPI Based BIOS Requirements

A serial flash device must meet the following minimum requirements when used explicitly for system BIOS storage.

- Erase size capability of at least one of the following: 64 Kbytes, 8 Kbytes, 4 Kbytes, or 256 bytes.
- Device must support multiple writes to a page without requiring a preceding erase cycle (See Section 4.21.6)
- Serial flash device must ignore the upper address bits such that an address of FFFFFFh aliases to the top of the flash memory.
- SPI Compatible Mode 0 support (clock phase is 0 and data is latched on the rising edge of the clock).
- If the device receives a command that is not supported or incomplete (less than 8 bits), the device must complete the cycle gracefully without any impact on the flash content.
- An erase command (page, sector, block, chip, etc.) must set all bits inside the designated area (page, sector, block, chip, etc.) to 1 (Fh).
- Status Register bit 0 must be set to 1 when a write, erase or write to status register is in progress and cleared to 0 when a write or erase is NOT in progress.
- Devices requiring the Write Enable command mst automatically clear the Write Enable Latch at the end of Data Program instructions.
- Byte write must be supported. The flexibility to perform a write between 1 byte to 64 bytes is recommended.
- Hardware Sequencing requirements are optional in BIOS only platforms.
- SPI flash parts that do not meet Hardware sequencing command set requirements may work in BIOS only platforms via software sequencing.

4.21.5.2 Intel® Management Engine Firmware SPI Flash Requirements

Intel® Management Engine Firmware must meet the SPI flash based BIOS Requirements plus:

- Hardware Sequencing.
- Flash part must be uniform 4 KB erasable block throughout the entire device or have 64 KB blocks with the first block (lowest address) divided into 4 KB or 8 KB blocks.
- Write protection scheme must meet SPI flash unlocking requirements for Management Engine.

4.21.5.2.1 SPI Flash Unlocking Requirements for Management Engine

Flash devices must be globally unlocked (read, write and erase access on the ME region) from power on by writing 00h to the flash's status register to disable write protection.

If the status register must be unprotected, it must use the enable write status register command 50h or write enable 06h.

Opcode 01h (write to status register) must then be used to write a single byte of 00h into the status register. This must unlock the entire part. If the SPI flash's status register has non-volatile bits that must be written to, bits [5:2] of the flash's status register must be all 0h to indicate that the flash is unlocked.

If bits [5:2] return a non zero values, the Intel® ME firmware will send a write of 00h to the status register. This must keep the flash part unlocked.

If there is no need to execute a write enable on the status register, then opcodes 06h and 50h must be ignored.

After global unlock, BIOS has the ability to lock down small sections of the flash as long as they do not involve the ME region.

4.21.5.3 Hardware Sequencing Requirements

The following table provides a list of commands and the associated opcodes that a SPI-based serial flash device must support in order to be compatible with hardware sequencing.

Table 4-60. Hardware Sequencing Commands and Opcode Requirements

Commands	Opcode	Notes
Write to Status Register	01h	Writes a byte to SPI flash's status register. Enable Write to Status Register command must be run prior to this command.
Program Data	02h	Single byte or 64 byte write as determined by flash part capabilities and software.
Read Data	03h	
Write Disable	04h	
Read Status	05h	Outputs contents of SPI flash's status register
Write Enable	06h	
Fast Read	0Bh	
Enable Write to Status Register	50h or 60h	Enables a bit in the status register to allow an update to the status register

Table 4-60. Hardware Sequencing Commands and Opcode Requirements

Commands	Opcode	Notes
Erase	Programmable	256B, 4 Kbyte, 8 Kbyte or 64 Kbyte
Full Chip Erase	C7h	
JEDEC ID	9Fh	See Section 4.21.5.3.1.

4.21.5.3.1 JEDEC ID

Since each serial flash device may have unique capabilities and commands, the JEDEC ID is the necessary mechanism for identifying the device so the uniqueness of the device can be comprehended by the controller (master). The JEDEC ID uses the opcode 9Fh and a specified implementation and usage model. This JEDEC Standard Manufacturer and Device ID read method is defined in Standard JESD21-C, PRN03-NV.

4.21.6 Multiple Page Write Usage Model

The system BIOS usage model require that the serial flash device support multiple writes to a page (minimum of 512 writes) without requiring a preceding erase command. BIOS commonly uses capabilities such as counters that are used for error logging and system boot progress logging. These counters are typically implemented by using byte-writes to 'increment' the bits within a page that have been designated as the counter.

Note: This usage model requirement is based on any given bit only being written once from a '1' to a '0'without requiring the preceding erase. An erase would be required to change bits back to the 1 state.

4.21.6.1 Soft Flash Protection

There are two types of flash protection that are not defined in the flash descriptor but supported by the system:

1. BIOS Range Write Protection
2. SMI#-Based Global Write Protection

Both mechanisms are logically OR'd together such that if any of the mechanisms indicate that the access should be blocked, then it is blocked. Table 4-61 provides a summary of the mechanisms.

Table 4-61. Flash Protection Mechanism Summary

Mechanism	Accesses Blocked	Range Specific?	Reset-Override or SMI#-Override?
BIOS Range Write Protection	Writes	Yes	Reset Override
Write Protect	Writes	No	SMI# Override

A blocked command will appear to software to finish, except that the Blocked Access status bit is set in this case.

4.21.6.2 BIOS Range Write Protection

The system provides a method for blocking writes to specific ranges in the SPI flash when the Protected BIOS Ranges are enabled. This is achieved by checking the Opcode type information (which can be locked down by the initial Boot BIOS) and the address of the requested command against the base and limit fields of a Write Protected BIOS range.

Note: Once BIOS has locked down the Protected BIOS Range registers, this mechanism remains in place until the next system reset.

4.21.6.3 SMI# Based Global Write Protection

The system provides a way to block writes to the SPI flash when the Write Protected bit is cleared (for example, protected). This is achieved by checking the Opcode type information (which can be locked down by the initial boot BIOS) of the requested command.

4.21.7 Flash Device Configurations

All platforms must have a SPI flash connected directly with a valid descriptor and Intel® Management Engine Firmware.

4.21.8 SPI Flash Device Recommended Pinout

The table below provides the recommended serial flash device pin-out for an 8-pin device. Using the recommended pin-out on an 8-pin device reduces complexities involved with designing the serial flash device onto a motherboard and allows for support of a common footprint usage model (see Section 4.21.9.1).

Table 4-62. Recommended Pinout for 8-Pin Serial Flash Device

Pin #	Signal
1	Chips Select
2	Data Output
3	Write Protect
4	Ground
5	Data Input
6	Serial Clock
7	Hold / Reset
8	Supply Voltage

Although an 8-pin device is preferred due to footprint compatibility, the following table contains the recommended serial flash device pinout for a 16-pin SOIC.

4.21.9 Serial Flash Device Package

Table 4-63. Recommended Pinout for 16-Pin Serial Flash Device

Pin #	Signal	Pin #	Signal
1	Hold / Reset	9	Write Protect
2	Supply Voltage	10	Ground
3	No Connect	11	No Connect
4	No Connect	12	No Connect
5	No Connect	13	No Connect
6	No Connect	14	No Connect
7	Chip Select	15	Serial Data In
8	Serial Data Out	16	Serial Clock

4.21.9.1 Common Footprint Usage Model

To minimize platform motherboard redesign and to enable platform Bill of Material (BOM) selectability, many PC System OEMs design their motherboard with a single common footprint. This common footprint allows population of a soldered down device or a socket that accepts a leadless device. This enables the board manufacturer to support, via selection of the appropriate BOM, either of these solutions on the same system without requiring any board redesign.

The common footprint usage model is desirable during system debug and by flash content developers since the leadless device can be easily removed and reprogrammed without damage to device leads. When the board and flash content is mature for high-volume production, both the socketed leadless solution and the soldered down leaded solution are available through BOM selection.

4.21.9.2 Serial Flash Device Package Recommendations

It is highly recommended that the common footprint usage model be supported. An example of how this can be accomplished is as follows:

- The recommended pinout for 8-pin serial flash devices is used (see Section 4.21.8).
- The 8-pin device is supported in either an 8-contact VDFPN (6x5 mm MLP) package or an 8-contact WSON (5x6 mm) package. These packages can fit into a socket that is land pattern compatible with the wide body SO8 package.
- The 8-pin device is supported in the SO8 (150 mil) and in the wide-body SO8 (200 mil) packages.

The 16-pin device is supported in the SO16 (300 mil) package.

4.22 Feature Capability Mechanism

A set of registers is included in the LPC Interface (Bus0:Device 31, Function 0, offset E0h - EBh) that allows the system software or BIOS to easily determine system supported features. These registers can be accessed through LPC PCI configuration space, thus allowing for convenient single point access mechanism for chipset feature detection.

This set of registers consists of:

- Capability ID (FDCAP)
- Capability Length (FDLEN)
- Capability Version
- Vendor-Specific Capability ID (FDVER) Feature Vector (FVECT)

§ §

5.0 Register and Memory Mappings

The system contains registers that are located in the processor's I/O space and memory space and sets of PCI configuration registers that are located in PCI configuration space. This chapter describes the I/O and memory maps at the register-set level. Register access is also described. Register-level address maps and individual register bit descriptions are provided in the following chapters.

5.1 I/O Map

The I/O map is divided into separate types. Fixed ranges cannot be moved, but in some cases can be disabled. Variable ranges can be moved and can also be disabled.

5.1.1 Fixed I/O Address Ranges

Table 5-2 shows the fixed I/O decode ranges from the CPU perspective. For each I/O range, there may be separate behavior for reads and writes. DMI cycles that go to target ranges that are marked as Reserved will not be decoded, and will be passed to the subtractive decode agent.

Address ranges that are not listed or marked RESERVED are NOT positively decoded by (unless assigned to one of the variable ranges). In subtractive mode, I/O ranges that are not otherwise decoded will be forwarded to PCIe*.

Table 5-2. Fixed I/O Ranges Decoded (Sheet 1 of 3)

I/O Address	Read Target	Write Target	Internal Unit	Separate Enable/Disable
00h – 08h	DMA Controller	DMA Controller	DMA	None
09h – 0Eh	RESERVED			
0Fh	DMA Controller	DMA Controller	DMA	None
10h – 18h	DMA Controller	DMA Controller	DMA	None
19h – 1Eh	RESERVED			
1Fh	DMA Controller	DMA Controller	DMA	None
20h – 21h	Interrupt Controller	Interrupt Controller	Interrupt	None
24h – 25h	Interrupt Controller	Interrupt Controller	Interrupt	None
28h – 29h	Interrupt Controller	Interrupt Controller	Interrupt	None
2Ch – 2Dh	Interrupt Controller	Interrupt Controller	Interrupt	None
2E-2F	LPC SIO	LPC SIO	Forwarded to LPC	Yes
30h – 31h	Interrupt Controller	Interrupt Controller	Interrupt	None
34h – 35h	Interrupt Controller	Interrupt Controller	Interrupt	None
38h – 39h	Interrupt Controller	Interrupt Controller	Interrupt	None
3Ch – 3Dh	Interrupt Controller	Interrupt Controller	Interrupt	None

Table 5-2. Fixed I/O Ranges Decoded (Sheet 2 of 3)

I/O Address	Read Target	Write Target	Internal Unit	Separate Enable/Disable
40h – 42h	Timer/Counter	Timer/Counter	PIT (8254)	None
43h	RESERVED			
4E-4F	LPC SIO	LPC SIO	Forwarded to LPC	Yes
50h – 52h	Timer/Counter	Timer/Counter	PIT	None
53h	RESERVED			
60h	Micocontroller	Microcontroller	Forwarded to LPC	Yes w/ 64h
61h	NMI Controller	NMI Controller	CPU I/F	None
62h	Microcontroller	Microcontroller	Forwarded to LPC	Yes w/ 66h
63h	NMI Controller[1]	NMI Controller[1]	CPU I/F	Yes, alias to 61h
64h	Micocontroller	Microcontroller	Forwarded to LPC	Yes w/ 60h
65h	NMI Controller[1]	NMI Controller[1]	CPU I/F	Yes, alias to 61h
66h	Microcontroller	Microcontroller	Forwarded to LPC	Yes w/ 62h
67h	NMI Controller[1]	NMI Controller[1]	CPU I/F	Yes, alias to 61h
70h	RESERVED[3]			
71h	RTC Controller	RTC Controller	RTC	None
72h	RTC Controller	NMI and RTC Controller	RTC	Yes, w/ 73h
73h	RTC Controller	RTC Controller	RTC	Yes, w/ 72h
74h	RTC Controller	NMI and RTC Controller	RTC	None
75h	RTC Controller	RTC Controller	RTC	None
76h	RTC Controller	NMI and RTC Controller	RTC	None
77h	RTC Controller	RTC Controller	RTC	None
80h	DMA Controller, or LPC, or PCI	DMA Controller, or LPC, or PCI	DMA	None
81h – 83h	DMA Controller	DMA Controller	DMA	None
84h - 86h	DMA Controller	DMA Controller and LPC or PCI	DMA	None
87h	DMA Controller	DMA Controller	DMA	None
88h	DMA Controller	DMA Controller and LPC or PCI	DMA	None
89h - 8Bh	DMA Controller	DMA Controller	DMA	None
8Ch - 8Eh	DMA Controller	DMA Controller and LPC or PCI	DMA	None
8Fh	DMA Controller	DMA Controller	DMA	None
90h - 91h	DMA Controller	DMA Controller	DMA	Yes, alias to 8xh
92h	Reset Generator	Reset Generator	CPU I/F	None
93h - 9Fh	DMA Controller	DMA Controller	DMA	Yes, alias to 8xh
A0h - A1h	Interrupt Controller	Interrupt Controller	Interrupt	None
A4h - A5h	Interrupt Controller	Interrupt Controller	Interrupt	None
A8h - A9h	Interrupt Controller	Interrupt Controller	Interrupt	None
ACh - ADh	Interrupt Controller	Interrupt Controller	Interrupt	None

Table 5-2. Fixed I/O Ranges Decoded (Sheet 3 of 3)

I/O Address	Read Target	Write Target	Internal Unit	Separate Enable/Disable
B0h - B1h	Interrupt Controller	Interrupt Controller	Interrupt	None
B2h - B3h	Power Management	Power Management	Power Management	None
B4h - B5h	Interrupt Controller	Interrupt Controller	Interrupt	None
B8h - B9h	Interrupt Controller	Interrupt Controller	Interrupt	None
BCh - BDh	Interrupt Controller	Interrupt Controller	Interrupt	None
C0h - D1h	DMA Controller	DMA Controller	DMA	None
D2h - DDh	RESERVED			
DEh - DFh	DMA Controller	DMA Controller	DMA	None
F0h	FERR#/IGNNE# / Interrupt Cont.	FERR#/IGNNE# / Interrupt Cont.	CPU I/F	None
170h - 177h	Reserved			
1F0h - 1F7h	Reserved			
200-207h	Gameport Low	Gameport Low	Forwarded to LPC	Yes
208-20Fh	Gameport High	Gameport High	Forwarded to LPC	Yes
376h	Reserved			
3F6h	Reserved			
4D0h – 4D1h	Interrupt Controller	Interrupt Controller	Interrupt	None
CF9h	Reset Generator	Reset Generator	CPU I/F	None

1. Only if the Port 61 Alias Enable bit (GCS.P61AE) bit is set.

5.1.2 Variable I/O Decode Ranges

Table 5-3 shows the variable I/O Decode ranges. They are set using Base Address Registers (BARs) or other config bits in the various configuration spaces. The PnP software (ACPI) can use their configuration mechanisms to set and adjust these values.

Warning: The variable I/O ranges should not be set to conflict with the fixed I/O ranges. Unpredictable results if the configuration software allows conflicts to occur.

Table 5-3. Variable I/O Decode Ranges (Sheet 1 of 2)

Range Name	Mappable	Size (Bytes)	Target
ACPI	Anywhere in 64K I/O Space	64	Power Management
IDE Bus Master	Anywhere in 64K I/O Space	1. 16 or 32 Bytes 2. 16 Bytes	1. IDE (SATA* Host Controller #1, #2)
SMBus	Anywhere in 64K I/O Space	32	SMBus Unit
TCO	96 bytes above ACPI base	32	TCO Unit
GPIO	Anywhere in 64K I/O space	128	GPIO Unit
Parallel Port	3 ranges in 64K I/O Space	8 (See Note 3)	LPC Peripheral
Serial Port 1	8 Ranges in 64K I/O Space	8	LPC Peripheral
Serial Port 2	8 Ranges in 64K I/O Space	8	LPC Peripheral

Table 5-3. Variable I/O Decode Ranges (Sheet 2 of 2)

Range Name	Mappable	Size (Bytes)	Target
Floppy Disk Controller	2 Ranges in 64K I/O Space	8	LPC Peripheral
LPC Generic 1	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC
LPC Generic 2	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC
LPC Generic 3	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC
LPC Generic 4	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC
IO Trapping Ranges	Anywhere in 64K I/O Space	1 to 256 Bytes	Trap on Backbone
Native IDE Cmd	Anywhere in 64K I/O Space[1]	8	IDE (SATA* Host Controller #2)
Native IDE Ctrl	Anywhere in 64K I/O Space[1]	4	IDE (SATA* Host Controller #2)
Serial ATA Index/Data Pair	Anywhere in 64K I/O Space	16	SATA* Host Controller #1, #2
PCI-Express Root Ports	Anywhere in 64K I/O Space	I/O Base/Limit	PCIe* Root Ports
Keyboard and Text (KT)	Anywhere in 64K I/O Space	8	Keyboard and Text

Notes:
1. All ranges are decoded directly from DMI.
2. There is also an alias 400h above the parallel port range that is used for ECP parallel ports.

5.2 Memory Map

Table 5-4 shows (from the CPU perspective) the memory ranges that will decode. Cycles that arrive from DMI that are not directed to any of the internal memory targets that decode directly from DMI (see Table 5-4) will be passed to the subtractive decode agent.

Table 5-4. Memory Decode Ranges (From CPU Perspective) (Sheet 1 of 2)

Memory Range	Target	Dependency/Comments
FECXX000 - FECXX040	I/O(x)APIC	X controlled via APIC Range Select (ASEL) field and APIC Enable (AEN) bit.
FEC10000 - FEC17FFF	PCIe* port 1	PCIe* root port 1 I/OxApic Enable (PAE) set
FEC18000 - FEC1FFFF	PCIe* port 2	PCIe* root port 2 I/OxApic Enable (PAE) set
FEC20000 - FEC27FFF	PCIe* port 3	PCIe* root port 3 I/OxApic Enable (PAE) set
FEC28000 - FEC2FFFF	PCIe* port 4	PCIe* root port 4 I/OxApic Enable (PAE) set
1 KB anywhere in 4GB range	USB2 Host Controller	Enable via standard PCI mechanism (Device 29, Function 0)
FED0 X000h-FED0 X3FFh	HPET	BIOS determines “fixed” location which is one of four 1KB ranges where X (in the first column) is 0h, 1h, 2h, or 3h.
FED4_0000h – FED4_BFFFh	TPM	TPM Memory Range on LPC.

Table 5-4. Memory Decode Ranges (From CPU Perspective) (Sheet 2 of 2)

Memory Range	Target	Dependency/Comments
64KB anywhere in 4GB range	LPC	LPC Generic Memory Range. Enable via setting bit[0] of the LPC Generic Memory Range register (B0:D31:F0:offset 98h).
32 Bytes anywhere in 64-bit address range	SMBus	Enable via standard PCI mechanism (Device 31: Function 3)
2KB anywhere above 64KB to 4GB range	SATA* Host Controller #1	AHCI memory-mapped registers. Enable via standard PCI mechanism (Device 31: Function 2)
Memory Base/Limit anywhere in 4GB range	PCI-Express Root Ports 1-4	Enable via standard PCI mechanism (Device 28: Function 0-3)
Prefetchable Memory Base/Limit anywhere in 64-bit address range	PCI-Express Root Ports 1-4	Enable via standard PCI mechanism (Device 28: Function 0-3)
4KB anywhere in 4GB range	Thermal Reporting	Enable via standard PCI mechanism (Device 31: Function 6)
16KB anywhere in 4GB range	Root Complex Register Block (RCRB)	Enable via setting bit[0] of the Root Complex Base Address register (B0:D31:F0:offset F0h).

5.3 Boot-Block Update Scheme

The system supports a "Top-Block Swap" mode that has the system swap the top block in the SPI (the boot block) with another location. This allows for safe update of the Boot Block (even if a power failure occurs). When the "top-swap" enable bit is set, the System will invert A16 for cycles going to the upper two blocks in the SPI. Specifically, in this mode accesses to FFFF_0000h-FFFF_FFFFh are directed to FFFE_0000h-FFFE_FFFFh and vice versa. When the Top Swap Enable bit is 0, the System will not invert A16. This bit is automatically set to 0 by RTEST#, but not by PLTRST#.

The scheme is based on the concept that the top block is reserved as the "boot" block, and the block immediately below the top block is reserved for doing boot-block updates.

The algorithm is:

1. Software copies the top block to the block immediately below the top
2. Software checks that the copied block is correct. This could be done by performing a checksum calculation.
3. Software sets the "Top-Block Swap" bit. This will invert A16 for cycles going to the SPI.
4. Software erases the top block
5. Software writes the new top block
6. Software checks the new top block
7. Software clears the top-block swap bit
8. Software sets the Top_Swap Lock-Down bit

If a power failure occurs at any point after step 3, the system will be able to boot from the copy of the boot block that is stored in the block below the top. This is because the top-swap bit is backed in the RTC well.

There is one remaining unusual case that could occur if the RTC battery is not sufficiently high to maintain the RTC well. To avoid the potentially fatal case (where the Top-Swap bit is NOT set, but the top block is not valid), a pinstrap will allow forcing the top-swap bit to be set. This would be a last resort to allow the user to get the system to boot (and avoid having to de-solder the SPI).

See the strapping section above for the signal used as the emergency strap.

When the top-swap strap is used, the top-swap bit will be forced to 1 (cannot be cleared by software). The algorithm to put in the BIOS spec is as follows:

1. If an RTC well power failure is experienced during a boot block update, the system will probably not be able to boot at that point.
2. The user can set the Top-Swap strap and force the system to boot from the second block. The code in the second block should read the valid BIOS image from disk (probably a floppy or CD-ROM) and put it into the top-swap.
3. The BIOS will not be able to clear the Top-Swap bit because the jumper is in place. The user should then remove the jumper and reboot.

§ §

6.0 LPC Interface Bridge Registers (B0:D31:F0)

6.1 Overview

The LPC bridge function of the PCH resides in PCI Bus 0:Device 31:Function 0. This function contains many other functional units, such as DMA and Interrupt controllers, Timers, Power Management, System Management, GPIO, RTC, and LPC Configuration Registers.

Registers and functions associated with other functional units (EHCI,... etc.) are described in their respective sections.

6.1.1 PCI Configuration Registers (LPC I/F—B0:D31:F0)

Note: Address locations that are not shown should be treated as Reserved.

Table 6-1. PCI Configuration Registers (LPC I/F—B0:D31:F0) (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: ID: Vendor Identification Register" on page 241	8086h
02h	03h	"Offset 02h: DID—Device Identification Register (LPC I/F—B0:D31:F0)" on page 241	2310h/2390h [1]
04h	05h	"Offset 04h: ID—PCICMD—PCI COMMAND Register (LPC I/F—B0:D31:F0)" on page 242	0007h
06h	07h	"Offset 06h: ID—PCISTS—PCI Status Register (LPC I/F—B0:D31:F0)" on page 243	0210h
08h	08h	"Offset 08h: RID—Revision Identification Register (PCI-PCI—B0:D31:F0)" on page 244	See register description
09h	09h	"Offset 09h: PI—Programming Interface Register (PCI-PCI—B0:D31:F0)" on page 244	00h
0Ah	09h	"Offset 0Ah: SCC—Sub Class Code Register (PCI-PCI—B0:D31:F0)" on page 244	01h
0Bh	0Bh	"Offset 0Bh: BCC—Base Class Code Register (LPC I/F—B0:D31:F0)" on page 245	06h
0Dh	0Dh	"Offset 0Dh: PLT—Primary Latency Timer Register (LPC I/F—B0:D31:F0)" on page 245	00h
0Eh	0Eh	"Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—B0:D31:F0)" on page 245	80h
2Ch	2Fh	"Offset 2Ch: SS—Sub System Identifiers Register (LPC I/F—B0:D31:F0)" on page 246	00000000h
40h	43h	"Offset 40h: PMBASE—ACPI Base Address Register (LPC I/F—B0:D31:F0)" on page 246	00000001h
44h	44h	"Offset 44h: ACPI_CNTL—ACPI Control Register (LPC I/F—B0:D31:F0)" on page 247	00h
48h	4Bh	"Offset 48h: GPIOBASE—GPIO Base Address Register (LPC I/F—B0:D31:F0)" on page 248	00000001h
4Ch	4Ch	"Offset 4Ch: GC—GPIO Control Register (LPC I/F—B0:D31:F0)" on page 249	00h
60h	63h	"Offset 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control Register (LPC I/F—B0:D31:F0)" on page 250	80h

Table 6-1. PCI Configuration Registers (LPC I/F—B0:D31:F0) (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
64h	64h	"Offset 64h: SIRQ_CNTL—Serial IRQ Control Register (LPC I/F—B0:D31:F0)" on page 251	10h
68h	6Bh	"Offset 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control Register (LPC I/F—B0:D31:F0)" on page 252	80h
6Ch	6Dh	"Offset 6Ch: LPC_IBDF—IOxAPIC Bus:Device:Function (LPC I/F—B0:D31:F0)" on page 253	00F8h
70h	7Fh	"Offset 70h: LPC_HnBDF—HPET n Bus:Device:Function (LPC I/F—B0:D31:F0)" on page 254	00F8h
80h	80h	"Offset 80h: LPC_I/O_DEC—I/O Decode Ranges Register (LPC I/F—B0:D31:F0)" on page 255	0000h
82h	83h	"Offset 82h: LPC_EN—LPC I/F Enables Register (LPC I/F—B0:D31:F0)" on page 256	0000h
84h	87h	"Offset 84h: GEN1_DEC—LPC I/F Generic Decode Range 1 Register (LPC I/F—B0:D31:F0)" on page 258	00000000h
88h	8Bh	"Offset 88h: GEN2_DEC—LPC I/F Generic Decode Range 2 Register (LPC I/F—B0:D31:F0)" on page 259	00000000h
8Ch	8Eh	"Offset 8Ch: GEN3_DEC—LPC I/F Generic Decode Range 3 Register (LPC I/F—B0:D31:F0)" on page 260	00000000h
90h	93h	"Offset 90h: GEN4_DEC—LPC I/F Generic Decode Range 4 Register (LPC I/F—B0:D31:F0)" on page 261	00000000h
94h	97h	"Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control (LPC I/F—B0:D31:F0)" on page 262	00002000h
98h	9Bh	"Offset 98h: LGMR—LPC I/F Generic Memory Range (LPC I/F—B0:D31:F0)" on page 264	00000000h
DCh	DCh	"Offset DCh: BIOS_CNTL—BIOS Control Register (LPC I/F—B0:D31:F0)" on page 265	20h
E0h	E1h	"Offset E0h: FDCAP—Feature Detection Capability ID (LPC I/F—B0:D31:F0)" on page 266	0009h
E2h	E2h	"Offset E2h: FDLEN—Feature Detection Capability Length (LPC I/F—B0:D31:F0)" on page 266	0Ch
E3h	E3h	"Offset E3h: FDVER—Feature Detection Version (LPC I/F—B0:D31:F0)" on page 267	10h
E4h	EBh	"Offset E4h: FDVCT—Feature Vector (LPC I/F—B0:D31:F0)" on page 267	See register description
F0h	F3h	"Offset F0h: RCBA—Root Complex Base Address Register (LPC I/F—B0:D31:F0)" on page 268	00000000h
A0h	A0h	"Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0)" on page 311	0000h
A2h	A2h	"Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0)" on page 313	00h
A4h	A4h	"Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0)" on page 315	00h
A6h	A6h	"Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register" on page 318	00h
A9h	A9h	"Offset A9h: Chipset Initialization Register 4 (PM—B0:D31:F0)" on page 318	01h
ABh	ABh	"Offset ABh: BM_BREAK_EN Register (PM—B0:D31:F0)" on page 319	01h
ACh	ACh	"Offset ACh: PMIR—Power Management Initialization Register (PM—B0:D31:F0)" on page 319	00000000h

Table 6-1. PCI Configuration Registers (LPC I/F—B0:D31:F0) (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
B8h	BBh	"Offset B8h: GPIO_ROUT—GPIO Routing Control Register (PM—B0:D31:F0)" on page 320	00000000h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

6.1.1.1 Offset 00h: VID—Vendor Identification Register (LPC I/F—B0:D31:F0)

Table 6-2. Offset 00h: ID: Vendor Identification Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification — This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

6.1.1.2 Offset 02h: DID—Device Identification Register (LPC I/F—B0:D31:F0)

Table 6-3. Offset 02h: DID—Device Identification Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2310h/2390h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device Identification — This is a 16-bit value assigned to the PCH's LPC bridge PCI device		2310h/2390h[1]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

6.1.1.3 Offset 04h: ID—PCICMD—PCI COMMAND Register (LPC I/F—B0:D31:F0)

Table 6-4. Offset 04h: ID—PCICMD—PCI COMMAND Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0007h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved		00h	RO
09	FBE	Fast Back to Back Enable — Hardwired to '0' as per *PCI Express* Specification*.		0h	RO
08	SEE	SERR# Enable: 0 = LPC bridge does not generates SERR# 1 = LPC bridge generates SERR#		0h	RW
07	WCC	Wait Cycle Control - Hardwired to '0' as per *PCI Express* Specification*.		0h	RO
06	PERE	Parity Error Response Enable: 0 = No action is taken when detecting a parity error. 1 = Enables the LPC bridge to respond to parity errors detected on ILB interface.		0h	RW
05	VGA_PSE	VGA Palette Snoop — Hardwired to '0' as per *PCI Express* Specification*.		0h	RO
04	MWIE	Memory Write and Invalidate Enable — Hardwired to '0'as per *PCI Express* Specification*.		0h	RO
03	SCE	Special Cycle Enable — Hardwired to '0' as per *PCI Express* Specification*.		0h	RO
02	BME	Bus Master Enable — Bus Masters cannot be disabled.		1	RO
01	MSE	Memory Space Enable — Memory space cannot be disabled on LPC.		1	RO
00	IOSE	I/O Space Enable — I/O space cannot be disabled on LPC.		1	RO

6.1.1.4 Offset 06h: ID—PCISTS—PCI Status Register (LPC I/F—B0:D31:F0)

Note: For the writable bits, software must write a 1 to clear bits that are set. Writing a 0 to the bit has no effect.

Table 6-5. Offset 06h: ID—PCISTS—PCI Status Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the Parity Error Response bit (D30:F0:04 bit 6) is not set.		0h	RWC
14	SSE	Signaled System Error — Set when the LPC bridge signals a system error to the internal SERR# logic.		0h	RWC
13	RMA	Received Master Abort: 0 = No master abort received. 1 = Set when the bridge receives a master abort status from the I/O data bus.		0h	RWC
12	RTA	Received Target Abort: 0 = No target abort received. 1 = Set when the bridge receives a target abort status from the I/O data bus.		0h	RWC
11	STA	Signaled Target Abort: 0 = No signaled target abort. 1 - Set when the bridge generates a completion packet with target abort status on the I/O data bus.		0h	RWC
10 :09	Reserved	Reserved		00h	RO
08	DPD	Data Parity Error Detected: 0 = Data parity error not detected. 1 = Set when the bridge receives a completion packet from the I/O data bus from a previous request, and detects a parity error, and CMD.PER is set (D30, F0, 04, bit 6).		0h	RWC
07 :05	Reserved	Reserved		0h	RO
04	CLIST	Capabilities List — Capability list exist on the PCI bridge.		1h	RO
03	IS	Interrupt Status — The PCI bridge does not generate interrupts.		0h	RO
02 :00	Reserved	Reserved		0h	RO

6.1.1.5 Offset 08h: RID—Revision Identification Register (LPC I/F—B0:D31:F0)

Table 6-6. Offset 08h: RID—Revision Identification Register (PCI-PCI—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: Indicates stepping of the host controller hardware		Variable	RO

6.1.1.6 Offset 09h: PI—Programming Interface Register (PCI-PCI—B0:D31:F0)

Table 6-7. Offset 09h: PI—Programming Interface Register (PCI-PCI—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PI	Programming Interface			RO

6.1.1.7 Offset 0Ah: SCC—Sub Class Code Register (LPC I/F—B0:D31:F0)

Table 6-8. Offset 0Ah: SCC—Sub Class Code Register (PCI-PCI—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0Ah **Offset End:** 09h	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code — 8-bit value that indicates the category of bridge for the LPC bridge. 01h = PCI-to-ISA bridge.			RO

6.1.1.8 Offset 0Bh: BCC—Base Class Code Register (LPC I/F—B0:D31:F0)

Table 6-9. Offset 0Bh: BCC—Base Class Code Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 06h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code — 8-bit value that indicates the type of device for the LPC bridge. 06h = Bridge device.			RO

6.1.1.9 Offset 0Dh: PLT—Primary Latency Timer Register (LPC I/F—B0:D31:F0)

Table 6-10. Offset 0Dh: PLT—Primary Latency Timer Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	MLC	Master Latency Count — Reserved			RO
02 :00	Reserved	Reserved			RO

6.1.1.10 Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—B0:D31:F0)

Table 6-11. Offset 0Eh: HEADTYP—Header Type Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 80h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Multi-Function Device — This bit is 1 to indicate a multi-function device.		1b	RO
06 :00	HEADTYP	Header Type — This 7-bit field identifies the header layout of the configuration space.		00h	RO

6.1.1.11 Offset 2Ch: SS—Sub System Identifiers Register (LPC I/F—B0:D31:F0)

This register is initialized to logic 0 by the assertion of PLTRST#. This register can be written only once after PLTRST# de-assertion.

Table 6-12. Offset 2Ch: SS—Sub System Identifiers Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2Ch **Offset End:** 2Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	SSID	Subsystem ID — This is written by BIOS. No hardware action taken on this value.		0000h	RWO
15 :00	SSVID	Subsystem Vendor ID — This is written by BIOS. No hardware action taken on this value.		0000h	RWO

6.1.1.12 Offset 40h: PMBASE—ACPI Base Address Register (LPC I/F—B0:D31:F0)

Sets base address for ACPI I/O registers, GPIO registers and TCO I/O registers. These registers can be mapped anywhere in the 64-K I/O space on 128-byte boundaries.

Table 6-13. Offset 40h: PMBASE—ACPI Base Address Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 40h **Offset End:** 43h	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved		0000h	RO
15 :07	BA	Base Address — This field provides 128 bytes of I/O space for ACPI, GPIO, and TCO logic. This is placed on a 128-byte boundary.		0000h	RW
06 :01	Reserved	Reserved		0000h	RO
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate I/O space.		1b	RO

6.1.1.13 Offset 44h: ACPI_CNTL—ACPI Control Register (LPC I/F—B0:D31:F0)

Table 6-14. Offset 44h: ACPI_CNTL—ACPI Control Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 44h **Offset End:** 44h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	ACPI_EN	ACPI Enable: 0 =Disable. 1 = Decode of the I/O range pointed to by the ACPI base register is enabled, and the ACPI power management function is enabled. NOTE: The APM power management ranges (B2/B3h) are always enabled and are not affected by this bit.		0b	RW
06 :03	Reserved	Reserved		0h	
02 :00	SCI_IRQ_SEL	SCI IRQ Select: Specifies on which IRQ the SCI will internally appear. If not using the APIC, the SCI must be routed to IRQ9–11, and that interrupt is not sharable with the SERIRQ stream, but is shareable with other PCI interrupts. If using the APIC, the SCI can also be mapped to IRQ20–23, and can be shared with other interrupts. **Bits** / **SCI Map** 000b: IRQ9 001b: IRQ10 010b: IRQ11 011b: Reserved 100b: IRQ20 (Only available if APIC enabled) 101b: IRQ21 (Only available if APIC enabled) 110b: IRQ22 (Only available if APIC enabled) 111b: IRQ23 (Only available if APIC enabled) When the interrupt is mapped to APIC interrupts 9, 10 or 11, the APIC should be programmed for active-high reception. When the interrupt is mapped to APIC interrupts 20 through 23, the APIC should be programmed for active-low reception.		00h	RW

6.1.1.14 Offset 48h: GPIOBASE—GPIO Base Address Register (LPC I/F—B0:D31:F0)

Table 6-15. Offset 48h: GPIOBASE—GPIO Base Address Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 48h **Offset End:** 4Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved. Always 0.			
15 :07	BA	Base Address — Provides the 128 bytes of I/O space for GPIO.			RW
06 :01	Reserved	Reserved. Always 0.			
00		Hardwired to 1 to indicate I/O space.			

6.1.1.15 Offset 4Ch: GC—GPIO Control Register (LPC I/F—B0:D31:F0)

Table 6-16. Offset 4Ch: GC—GPIO Control Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 4Ch **Offset End:** 4Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05	Reserved	Reserved			
04	EN	GPIO Enable — This bit enables/disables decode of the I/O range pointed to by the GPIO Base Address register (B0:D31:F0:48h) and enables the GPIO function. 0 = Disable. 1 = Enable.			RW
03 :01	Reserved	Reserved			
00	GLE	GPIO Lockdown Enable — This bit enables lockdown of the following GPIO registers: Offset 00h: GPIO_USE_SEL Offset 04h: GP_IO_SEL Offset 0Ch: GP_LVL Offset 30h: GPIO_USE_SEL2 Offset 34h: GP_IO_SEL2 Offset 38h: GP_LVL2 Offset 40h: GPIO_USE_SEL3 Offset 44h: GP_IO_SEL3 Offset 48h: GP_LVL3 Offset 60h: GP_RST_SEL 0 = Disable. 1 = Enable. When this bit is written from a '1' to a '0' an SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after they are locked down.			RW

6.1.1.16 Offset 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control Register (LPC I/F—B0:D31:F0)

Offset Address:

PIRQA – 60h
PIRQB – 61h
PIRQC – 62h
PIRQC - 63h

Table 6-17. Offset 60h: PIRQ[n]_ROUT—PIRQ[A,B,C,D] Routing Control Register (LPC I/F—B0:D31:F0)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D31:F0</td><td>Offset Start: 60h
Offset End: 63h</td></tr>
<tr><td colspan="2">Size: 8 bit</td><td colspan="2">Default: 80h</td><td></td><td>Power Well: Core</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>07</td><td>IRQEN</td><td>Interrupt Routing Enable
0 = The corresponding PIRQ is routed to one of the ISA-compatible interrupts specified in bits[3:0].
1 = The PIRQ is not routed to the 8259.

BIOS must program this bit to 0 during POST for any of the PIRQs that are being used. The value of this bit may subsequently be changed by the OS when setting up for I/O APIC interrupt delivery mode.</td><td></td><td></td><td>RW</td></tr>
<tr><td>06 :04</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>03 :00</td><td></td><td>IRQ Routing — (ISA compatible)

<table><tr><th>Value</th><th>IRQ</th><th></th><th>Value</th><th>IRQ</th></tr><tr><td>0000b</td><td>Reserved</td><td></td><td>1000b</td><td>Reserved</td></tr><tr><td>0001b</td><td>Reserved</td><td></td><td>1001b</td><td>IRQ9</td></tr><tr><td>0010b</td><td>Reserved</td><td></td><td>1010b</td><td>IRQ10</td></tr><tr><td>0011b</td><td>IRQ3</td><td></td><td>1011b</td><td>IRQ11</td></tr><tr><td>0100b</td><td>IRQ4</td><td></td><td>1100b</td><td>IRQ12</td></tr><tr><td>0101b</td><td>IRQ5</td><td></td><td>1101b</td><td>Reserved</td></tr><tr><td>0110b</td><td>IRQ6</td><td></td><td>1110b</td><td>IRQ14</td></tr><tr><td>0111b</td><td>IRQ7</td><td></td><td>1111b</td><td>IRQ15</td></tr></table></td><td></td><td></td><td>RW</td></tr>
</table>

6.1.1.17 Offset 64h: SIRQ_CNTL—Serial IRQ Control Register (LPC I/F—B0:D31:F0)

Table 6-18. Offset 64h: SIRQ_CNTL—Serial IRQ Control Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 64h **Offset End:** 64h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	SIRQEN	Serial IRQ Enable: The buffer is input only and internally SERIRQ will be a 1. 1 = Serial IRQs will be recognized. The SERIRQ pin will be configured as SERIRQ.			RW
06	SIRQMD	Serial IRQ Mode Select: The serial IRQ machine will be in quiet mode when this bit is 0. The serial IRQ machine will be in continuous mode when this bit is 1. For systems using Quiet Mode, this bit should be set to 1 (Continuous Mode) for at least one frame after coming out of reset before switching back to Quiet Mode. Failure to do so results in the PCH not recognizing SERIRQ interrupts.			RW
05 :02	SIRQSZ	Serial IRQ Frame Size — Fixed field that indicates the size of the SERIRQ frame as 21 frames.		4h	RO
01 :00	SFPW	Start Frame Pulse Width — This is the number of PCI clocks that the SERIRQ pin will be driven low by the serial IRQ machine to signal a start frame. In continuous mode, the PCH will drive the start frame for the number of clocks specified. In quiet mode, the PCH will drive the start frame for the number of clocks specified minus one, as the first clock was driven by the peripheral. 00 = 4 clocks 01 = 6 clocks 10 = 8 clocks 11 = Reserved			RW

6.1.1.18 Offset 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control Register (LPC I/F—B0:D31:F0)

Offset Address:

PIRQE – 68h
PIRQF – 69h
PIRQG – 6Ah
PIRQH - 6Bh

Table 6-19. Offset 68h: PIRQ[n]_ROUT—PIRQ[E,F,G,H] Routing Control Register (LPC I/F—B0:D31:F0)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D31:F0</td><td>Offset Start: 68h
Offset End: 6Bh</td></tr>
<tr><td colspan="2">Size: 8 bit</td><td colspan="2">Default: 80h</td><td></td><td>Power Well: Core</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>07</td><td>IRQEN</td><td>Interrupt Routing Enable:
The corresponding PIRQ is routed to one of the ISA-compatible interrupts specified in bits[3:0].
The PIRQ is not routed to the 8259.
BIOS must program this bit to 0 during POST for any of the PIRQs that are being used. The value of this bit may subsequently be changed by the OS when setting up for I/O APIC interrupt delivery mode.</td><td></td><td></td><td>RW</td></tr>
<tr><td>06 :04</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>03 :00</td><td></td><td>IRQ Routing — (ISA compatible.)
<table><tr><th>Value</th><th>IRQ</th><th></th><th>Value</th><th>IRQ</th></tr><tr><td>0000b</td><td>Reserved</td><td></td><td>1000b</td><td>Reserved</td></tr><tr><td>0001b</td><td>Reserved</td><td></td><td>1001b</td><td>IRQ9</td></tr><tr><td>0010b</td><td>Reserved</td><td></td><td>1010b</td><td>IRQ10</td></tr><tr><td>0011b</td><td>IRQ3</td><td></td><td>1011b</td><td>IRQ11</td></tr><tr><td>0100b</td><td>IRQ4</td><td></td><td>1100b</td><td>IRQ12</td></tr><tr><td>0101b</td><td>IRQ5</td><td></td><td>1101b</td><td>Reserved</td></tr><tr><td>0110b</td><td>IRQ6</td><td></td><td>1110b</td><td>IRQ14</td></tr><tr><td>0111b</td><td>IRQ7</td><td></td><td>1111b</td><td>IRQ15</td></tr></table></td><td></td><td></td><td>RW</td></tr>
</table>

6.1.1.19 Offset 6Ch: LPC_IBDF—IOxAPIC Bus:Device:Function (LPC I/F—B0:D31:F0)

Table 6-20. Offset 6Ch: LPC_IBDF—IOxAPIC Bus:Device:Function (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 6Ch **Offset End:** 6Dh	
Size: 16 bit	**Default:** 00F8h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	IBDF	IOxAPIC Bus:Device:Function — This field specifies the bus:device:function that PCH's IOxAPIC will be using for the following: As the Requester ID when initiating Interrupt Messages to the processor. As the Completer ID when responding to the reads targeting the IOxAPIC's Memory-Mapped I/O registers. The 16-bit field comprises the following: **Bits** / **Description** 15:8 — Bus Number 7:3 — Device Number 2:0 — Function Number This field defaults to Bus 0: Device 31: Function 0 after reset. BIOS can program this field to provide a unique bus:device:function number for the internal IOxAPIC.			RW

6.1.1.20 Offset 70h: LPC_HnBDF—HPET n Bus:Device:Function (LPC I/F—B0:D31:F0)

Offset:

H0BDF 70h-71h
H1BDF 72h-73h
H2BDF 74h-75h
H3BDF 76h-77h
H4BDF 78h-79h
H5BDF 7Ah-7Bh
H6BDF 7Ch-7Dh
H7BDF 7Eh-7Fh

Table 6-21. Offset 70h: LPC_HnBDF—HPET n Bus:Device:Function (LPC I/F—B0:D31:F0)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D31 :F0</td><td colspan="2">Offset Start: 70h
Offset End: 7Fh</td></tr>
<tr><td>Size: 16 bit</td><td colspan="2">Default: 00F8h</td><td></td><td colspan="2">Power Well: Core</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>15 :00</td><td>HnBDF</td><td>HPET n Bus:Device:Function — This field specifies the bus:device:function that the PCH’s HPET n will be using in the following:
-As the Requester ID when initiating Interrupt Messages to the CPU
-As the Completer ID when responding to the reads targeting the corresponding HPET’s Memory-Mapped I/O registers.

The 16-bit field comprises the following:

<table><tr><td>Bits</td><td>Description</td></tr><tr><td>15:08</td><td>Bus Number</td></tr><tr><td>07:03</td><td>Device Number</td></tr><tr><td>02:00</td><td>Function Number</td></tr></table>
This field is default to Bus 0: Device 31: Function 0 after reset. BIOS shall program this field accordingly if unique bus:device:function number is required for the corresponding HPET.</td><td></td><td></td><td>RW</td></tr>
</table>

6.1.1.21 Offset 80h: LPC_I/O_DEC—I/O Decode Ranges Register (LPC I/F—B0:D31:F0)

Table 6-22. Offset 80h: LPC_I/O_DEC—I/O Decode Ranges Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 80h **Offset End:** 80h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved			
12		FDD Decode Range — Determines which range to decode for the FDD Port 3F0h – 3F5h, 3F7h (Primary) 1 = 370h – 375h, 377h (Secondary)			RW
11 :10	Reserved	Reserved			
09 :08		LPT Decode Range — This field determines which range to decode for the LPT Port. 00 = 378h – 37Fh and 778h – 77Fh 01 = 278h – 27Fh (port 279h is read only) and 678h – 67Fh 10 = 3BCh –3BEh and 7BCh – 7BEh 11 = Reserved			RW
07	Reserved	Reserved			
06 :04		COMB Decode Range — This field determines which range to decode for the COMB Port. 000 = 3F8h – 3FFh (COM1) 001 = 2F8h – 2FFh (COM2) 010 = 220h – 227h 011 = 228h – 22Fh 100 = 238h – 23Fh 101 = 2E8h – 2EFh (COM4) 110 = 338h – 33Fh 111 = 3E8h – 3EFh (COM3)			RW
03	Reserved	Reserved			
02 :00		COMA Decode Range — This field determines which range to decode for the COMA Port. 000 = 3F8h – 3FFh (COM1) 001 = 2F8h – 2FFh (COM2) 010 = 220h – 227h 011 = 228h – 22Fh 100 = 238h – 23Fh 101 = 2E8h – 2EFh (COM4) 110 = 338h – 33Fh 111 = 3E8h – 3EFh (COM3)			RW

6.1.1.22 Offset 82h: LPC_EN—LPC I/F Enables Register (LPC I/F—B0:D31:F0)

Table 6-23. Offset 82h: LPC_EN—LPC I/F Enables Register (LPC I/F—B0:D31:F0) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13	CNF2_LPC_EN	Microcontroller Enable # 2.: 0 = Disable. 1 = Enables the decoding of the I/O locations 4Eh and 4Fh to the LPC interface. This range is used for a microcontroller.			RW
12	CNF1_LPC_EN	Super I/O Enable: 0 = Disable. 1 = Enables the decoding of the I/O locations 2Eh and 2Fh to the LPC interface. This range is used for Super I/O devices.			RW
11	MC_LPC_EN	Microcontroller Enable # 1.: 0 = Disable. 1 = Enables the decoding of the I/O locations 62h and 66h to the LPC interface. This range is used for a microcontroller.			RW
10	KBC_LPC_E	Keyboard Enable: 0 = Disable. 1 = Enables the decoding of the I/O locations 60h and 64h to the LPC interface. This range is used for a microcontroller.			RW
09	GAMEH_LPC_EN	High Gameport Enable: 0 = Disable. 1 = Enables the decoding of the I/O locations 208h to 20Fh to the LPC interface. This range is used for a gameport.			RW
08	GAMEL_LPC_E	Low Gameport Enable: 0 = Disable. 1 = Enables the decoding of the I/O locations 200h to 207h to the LPC interface. This range is used for a gameport.			RW
07 :04	Reserved	Reserved			
03	FDD_LPC_EN	Floppy Drive Enable 0 = Disable. 1 = Enables the decoding of the FDD range to the LPC interface. This range is selected in the LPC_FDD/LPT Decode Range Register (B0:D31:F0:80h, bit 12).			RW

Table 6-23. Offset 82h: LPC_EN—LPC I/F Enables Register (LPC I/F—B0:D31:F0) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	LPT_LPC_EN	Parallel Port Enable 0 = Disable. 1 = Enables the decoding of the LPT range to the LPC interface. This range is selected in the LPC_FDD/LPT Decode Range Register (B0:D31:F0:80h, bit 9:8).			RW
01	COMB_LPC_EN	Comm Port B Enable 0 = Disable. 1 = Enables the decoding of the COMB range to the LPC interface. This range is selected in the LPC_COM Decode Range Register (B0:D31:F0:80h, bits 6:4).			RW
00	COMA_LPC_EN	Comm Port A Enable 0 = Disable. 1 = Enables the decoding of the COMA range to the LPC interface. This range is selected in the LPC_COM Decode Range Register (B0:D31:F0:80h, bits 3:2).			RW

6.1.1.23 Offset 84h: GEN1_DEC—LPC I/F Generic Decode Range 1 Register (LPC I/F—B0:D31:F0)

Table 6-24. Offset 84h: GEN1_DEC—LPC I/F Generic Decode Range 1 Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :18	GEN1_MASK	Generic I/O Decode Range Address[7:2] Mask — A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.			RW
17 :16	Reserved	Reserved			
15 :02	GEN1_BASE	Generic I/O Decode Range 1 Base Address — This address is aligned on a 128-byte boundary, and must have address lines 31:16 as 0. Note: The PCH does not provide decode down to the word or byte level			RW
01	Reserved	Reserved			
00	GEN1_EN	Generic Decode Range 1 Enable 0 = Disable. 1 = Enable the GEN1 I/O range to be forwarded to the LPC I/F			RW

6.1.1.24 Offset 88h: GEN2_DEC—LPC I/F Generic Decode Range 2 Register (LPC I/F—B0:D31:F0)

Table 6-25. Offset 88h: GEN2_DEC—LPC I/F Generic Decode Range 2 Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 88h **Offset End:** 8Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :18	GEN2_MASK	Generic I/O Decode Range Address[7:2] Mask — A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.			RW
17 :16	Reserved	Reserved			
15 :02	GEN2_BASE	Generic I/O Decode Range 2 Base Address — The PCH does not provide decode down to the word or byte level			RW
01	Reserved	Reserved			
00	GEN2_EN	Generic Decode Range 2 Enable 0 = Disable. 1 = Enable the GEN2 I/O range to be forwarded to the LPC I/F			RW

6.1.1.25 Offset 8Ch: GEN3_DEC—LPC I/F Generic Decode Range 3 Register (LPC I/F—B0:D31:F0)

Table 6-26. Offset 8Ch: GEN3_DEC—LPC I/F Generic Decode Range 3 Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 8Ch **Offset End:** 8Eh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :18	GEN3_MASK	Generic I/O Decode Range Address[7:2] Mask — A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.			RW
17 :16	Reserved	Reserved			
15 :02	GEN3_BASE	Generic I/O Decode Range 3 Base Address — The PCH does not provide decode down to the word or byte level			RW
01	Reserved	Reserved			
00	GEN4_EN	Generic Decode Range 4 Enable: 0 = Disable. 1 = Enable the GEN3 I/O range to be forwarded to the LPC I/F			RW

6.1.1.26 Offset 90h: GEN4_DEC—LPC I/F Generic Decode Range 4 Register (LPC I/F—B0:D31:F0)

Table 6-27. Offset 90h: GEN4_DEC—LPC I/F Generic Decode Range 4 Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 90h **Offset End:** 93h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :18	GEN4_MASK	Generic I/O Decode Range Address[7:2] Mask — A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.			RW
17 :16	Reserved	Reserved			
15 :02	GEN4_BASE	Generic I/O Decode Range 4 Base Address — The PCH does not provide decode down to the word or byte level			RW
01	Reserved	Reserved			
00	GEN4_EN	Generic Decode Range 4 Enable: 0 = Disable. 1 = Enable the GEN3 I/O range to be forwarded to the LPC I/F			RW

6.1.1.27 Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control (LPC I/F—B0:D31:F0)

Table 6-28. Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control (LPC I/F—B0:D31:F0) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00002000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	SMIBYENDPS	SMI Caused by End of Pass-Through — This bit indicates if the event occurred. Even if the corresponding enable bit is not set in bit 7, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a 1 to the bit location in any of the controllers. 1 = Event Occurred			RWC
14 :12	Reserved	Reserved			
11	TRAPBY64W	SMI Caused by Port 64 Write — This bit indicates if the event occurred. Even if the corresponding enable bit is not set in bit 3, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. The A20Gate Pass-Through Logic allows specific port 64h writes to complete without setting this bit. 0 = Software clears this bit by writing a 1 to the bit location in any of the controllers. 1 = Event Occurred.			RWC
10	TRAPBY64R	SMI Caused by Port 64 Read — This bit indicates if the event occurred. Even if the corresponding enable bit is not set in bit 2, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a 1 to the bit location in any of the controllers. 1 = Event Occurred.			RWC
09	TRAPBY60W	SMI Caused by Port 60 Write — This bit indicates if the event occurred. Even if the corresponding enable bit is not set in bit 1, this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. The A20Gate Pass-Through Logic allows specific port 64h writes to complete without setting this bit. 0 = Software clears this bit by writing a 1 to the bit location in any of the controllers. 1 = Event Occurred.			RWC
08	TRAPBY60R	SMI Caused by Port 60 Read — This bit indicates if the event occurred. Even if the corresponding enable bit is not set in the bit 0, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. 0 = Software clears this bit by writing a 1 to the bit location in any of the controllers. 1 = Event Occurred.			RWC
07	SMIATENDPS	SMI at End of Pass-Through Enable — This bit enables SMI at the end of a pass-through. This can occur if an SMI is generated in the middle of a pass-through, and needs to be serviced later. 0 = Disable 1 = Enable			RW

Table 6-28. Offset 94h: ULKMC—USB Legacy Keyboard / Mouse Control (LPC I/F—B0:D31:F0) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00002000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	PSTATE	Pass Through State: 0 = If software needs to reset this bit, it should set bit 5 in all of the host controllers to 0. 1 = Indicates that the state machine is in the middle of an A20GATE pass-through sequence.			RO
05	A20PASSEN	A20Gate Pass-Through Enable: 0 = Disable. 1 = Enable. Allows A20GATE sequence Pass-Through function. A specific cycle sequence involving writes to port 60h and 64h does not result in the setting of the SMI status bits.			RW
04	USBSMIEN	SMI on USB IRQ Enable: 0 = Disable 1 = Enable. USB interrupt will cause an SMI event.			RW
03	64WEN	SMI on Port 64 Writes Enable: 0 = Disable 1 = Enable. A 1 in bit 11 will cause an SMI event.			RW
02	64REN	SMI on Port 64 Reads Enable: 0 = Disable 1= Enable. A 1 in bit 10 will cause an SMI event.			RW
01	60WEN	SMI on Port 60 Writes Enable: 0 = Disable 1 = Enable. A 1 in bit 9 will cause an SMI event.			RW
00	60REN	SMI on Port 60 Reads Enable: 0 = Disable 1 = Enable. A 1 in bit 8 will cause an SMI event.			RW

6.1.1.28 Offset 98h: LGMR—LPC I/F Generic Memory Range (LPC I/F—B0:D31:F0)

Table 6-29. Offset 98h: LGMR—LPC I/F Generic Memory Range (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		Memory Address [31:16] — This field specifies a 64 KB memory block anywhere in the 4 GB memory space that will be decoded to LPC as standard LPC memory cycle if enabled.			RW
15 :01	Reserved	Reserved			
00		LPC Memory Range Decode Enable — When this bit is set to '1', then the range specified in bits 31:16 of this register is enabled for decoding to LPC			RW

6.1.1.29 Offset DCh: BIOS_CNTL—BIOS Control Register (LPC I/F—B0:D31:F0)

Table 6-30. Offset DCh: BIOS_CNTL—BIOS Control Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** DCh **Offset End:** DCh	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 06	Reserved	Reserved			
05	SMM_BWP	SMM BIOS write protect disable — This bit set defines when the BIOS region can be written by the host. BIOS region SMM protection is disabled. The BIOS Region is writable regardless if Processors are in SMM or not. (Set this field to '0' for legacy behavior) BIOS region SMM protection is enabled. The BIOS Region is not writable unless all Processors are in SMM.			RWLO
04	TSS	Top Swap Status — This bit provides a read-only path to view the state of the Top Swap bit that is at offset 3414h, bit 0.			
03 :02	SRO	SPI Read Configuration — This 2-bit field controls two policies related to BIOS reads on the SPI interface: Bit 3- Prefetch Enable Bit 2- Cache Disable Settings are summarized as follows: **Bits 3:2 Description** **00b** No prefetching, but caching enabled. 64B demand reads load the read buffer cache with "valid" data, allowing repeated code fetches to the same line to complete quickly **01b** No prefetching and no caching. One-to-one correspondence of host BIOS reads to SPI cycles. This value can be used to invalidate the cache. **10b** Prefetching and Caching enabled. This mode is used for long sequences of short reads to consecutive addresses (for example, shadowing). **11b** Reserved. This is an invalid configuration, caching must be enabled when prefetching is enabled.			RW
01	BLE	BIOS Lock Enable: Setting the BIOSWE will not cause SMIs. Enables setting the BIOSWE bit to cause SMIs. Once set, this bit can only be cleared by a PLTRST#			RWLO
00	BIOSWE	BIOS Write Enable: Only read cycles result in Firmware Hub I/F cycles. Access to the BIOS space is enabled for both read and write cycles. When this bit is written from a 0 to a 1 and BIOS Lock Enable (BLE) is also set, an SMI# is generated. This ensures that only SMI code can update BIOS.			RW

6.1.1.30 Offset E0h: FDCAP—Feature Detection Capability ID (LPC I/F—B0:D31:F0)

Table 6-31. Offset E0h: FDCAP—Feature Detection Capability ID (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** E0h **Offset End:** E1h	
Size: 16 bit	**Default:** 0009h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Item Pointer — Configuration offset of the next Capability Item. 00h indicates the last item in the Capability List.			RO
07 :00		Capability ID — Indicates a Vendor Specific Capability			RO

6.1.1.31 Offset E2h: FDLEN—Feature Detection Capability Length (LPC I/F—B0:D31:F0)

Table 6-32. Offset E2h: FDLEN—Feature Detection Capability Length (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** E2h **Offset End:** E2h	
Size: 8 bit	**Default:** 0Ch			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	FDLEN	Capability Length — Indicates the length of this Vendor Specific capability, as required by PCI Spec.			RO

6.1.1.32 Offset E3h: FDVER—Feature Detection Version (LPC I/F—B0:D31:F0)

Table 6-33. Offset E3h: FDVER—Feature Detection Version (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** E3h **Offset End:** E3h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04		Vendor-Specific Capability ID — A value of 1h in this 4-bit field identifies this Capability as Feature Detection Type. This field allows software to differentiate the Feature Detection Capability from other Vendor-Specific capabilities			RO
03 :00		Capability Version — This field indicates the version of the Feature Detection capability			RO

6.1.1.33 Offset E4h: FDVCT—Feature Vector (LPC I/F—B0:D31:F0)

Table 6-34. Offset E4h: FDVCT—Feature Vector (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** E4h **Offset End:** EBh	
Size: 64 bit	**Default:** See register description			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :00	Reserved	Reserved			

6.1.1.34 Offset F0h: RCBA—Root Base Address Register (LPC I/F—B0:D31:F0)

Table 6-35. Offset F0h: RCBA—Root Complex Base Address Register (LPC I/F—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** F0h **Offset End:** F3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	BA	Base Address — Base Address for the root complex register block decode range. This address is aligned on a 16-KB boundary.			RW
13 :01	Reserved	Reserved			
00	EN	Enable — When set, enables the range specified in BA to be claimed as the Root Complex Register Block			RW

6.1.2 DMA I/O Registers (LPC I/F—B0:D31:F0)

Table 6-36. DMA I/O Registers (LPC I/F—B0:D31:F0)

Offset Start	**Offset End**	**Register ID - Description**	**Default Value**
00h	CCh	"Offset 00h: DMABASE_CA—DMA Base and Current Address Registers (LPC I/F—B0:D31:F0)" on page 269	Undefined
01h	CEh	"Offset 01h: DMABASE_CC—DMA Base and Current Count Registers (LPC I/F—B0:D31:F0)" on page 270	Undefined
87h	8Ah	"Offset 87h: DMAMEM_LP—DMA Memory Low Page Registers (LPC I/F—B0:D31:F0)" on page 271	Undefined
Ch.#0 - 3 = 08h	Ch.#4 - 7 = D0h	"Offset 08h: DMACMD—DMA Command Register (LPC I/F—B0:D31:F0)" on page 272	Undefined
Ch.#0 - 3 = 08h	Ch.#4 - 7 = D0h	"Offset 08h: DMASTA—DMA Status Register (LPC I/F—B0:D31:F0)" on page 273	Undefined
Ch.#0 - 3 = 0Ah	Ch.#4 - 7 = D4h	"Offset 0Ah: DMA_WRSMSK—DMA Write Single Mask Register (LPC I/F—B0:D31:F0)" on page 274	0000 01xx
Ch.#0 - 3 = 0Bh	Ch.#4 - 7 = D6h	"Offset 08h: DMACH_MODE—DMA Channel Mode Register (LPC I/F—B0:D31:F0)" on page 275	0000 00xx
Ch.#0 - 3 = 0Ch	Ch.#4 - 7 = D8h	"Offset 0Ch: DMA Clear Byte Pointer Register (LPC I/F—B0:D31:F0)" on page 276	XXXX XXXX
Ch.#0 - 3 = 0Ch	Ch.#4 - 7 = DAh	"Offset 0Dh: DMA Master Clear Register (LPC I/F—B0:D31:F0)" on page 276	XXXX XXXX
Ch.#0 - 3 = 0Eh	Ch.#4 - 7 = DCh	"Offset 0Eh: DMA Master Clear Register (LPC I/F—B0:D31:F0)" on page 276	XXXX XXXX
Ch.#0 - 3 = 0Fh	Ch.#4 - 7 = DEh	"Offset 0Fh: DMA_WRMSK—DMA Write All Mask Register (LPC I/F—B0:D31:F0)" on page 277	0000 1111

6.1.2.1 Offset 00h: DMABASE_CA—DMA Base and Current Address Registers (LPC I/F—B0:D31:F0)

I/O Address:

Ch. #0 = 00h
Ch. #1 = 02h
Ch. #2 = 04h
Ch. #3 = 06h
Ch. #5 = C4h
Ch. #6 = C8h
Ch. #7 = CCh

Table 6-37. Offset 00h: DMABASE_CA—DMA Base and Current Address Registers (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 00h **Offset End:** CCh	
Size: 16 bit[1]	**Default:** Undefined			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Base and Current Address — This register determines the address for the transfers to be performed. The address specified points to two separate registers. On writes, the value is stored in the *Base Address* register and copied to the *Current Address* register. On reads, the value is returned from the *Current Address* register. The address increments/decrements in the Current Address register after each transfer, depending on the mode of the transfer. If the channel is in auto-initialize mode, the Current Address register will be reloaded from the Base Address register after a terminal count is generated. For transfers to/from a 16-bit slave (channel's 5-7), the address is shifted left one bit location. Bit 15 will be shifted into Bit 16. The register is accessed in 8 bit quantities. The byte is pointed to by the current byte pointer flip/flop. Before accessing an address register, the byte pointer flip/flop should be cleared to ensure that the low byte is accessed first			RW

1. per channel but accessed in two 8-bit quantities

6.1.2.2 Offset 01h: DMABASE_CC—DMA Base and Current Count Registers (LPC I/F—B0:D31:F0)

I/O Address:

Ch. #0 = 01h
Ch. #1 = 03h
Ch. #2 = 05h
Ch. #3 = 07h
Ch. #5 = C6h
Ch. #6 = CAh
Ch. #7 = CEh

Table 6-38. Offset 01h: DMABASE_CC—DMA Base and Current Count Registers (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 01h **Offset End:** CEh	
Size: 16 bit[1]	**Default:** Undefined			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Base and Current Count — This register determines the number of transfers to be performed. The address specified points to two separate registers. On writes, the value is stored in the *Base Count* register and copied to the *Current Count* register. On reads, the value is returned from the *Current Count* register. The actual number of transfers is one more than the number programmed in the Base Count Register (for example, programming a count of 4h results in 5 transfers). The count is decrements in the Current Count register after each transfer. When the value in the register rolls from 0 to FFFFh, a terminal count is generated. If the channel is in auto-initialize mode, the Current Count register will be reloaded from the Base Count register after a terminal count is generated. For transfers to/from an 8-bit slave (channels 0–3), the count register indicates the number of bytes to be transferred. For transfers to/from a 16-bit slave (channels 5–7), the count register indicates the number of words to be transferred. The register is accessed in 8 bit quantities. The byte is pointed to by the current byte pointer flip/flop. Before accessing a count register, the byte pointer flip/flop should be cleared to ensure that the low byte is accessed first.			RW

1.per channel but accessed in two 8-bit quantities

6.1.2.3 Offset 87h: DMAMEM_LP—DMA Memory Low Page Registers (LPC I/F—B0:D31:F0)

I/O Address:

Ch. #0 = 87h
Ch. #1 = 83h
Ch. #2 = 81h
Ch. #3 = 82h
Ch. #5 = 8Bh
Ch. #6 = 89h
Ch. #7 = 8Ah

Table 6-39. Offset 87h: DMAMEM_LP—DMA Memory Low Page Registers (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 87h **Offset End:** 8Ah	
Size: 8 bit	**Default:** Undefined			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		DMA Low Page (ISA Address bits [23:16]) — This register works in conjunction with the DMA controller's Current Address Register to define the complete 24-bit address for the DMA channel. This register remains static throughout the DMA transfer. Bit 16 of this register is ignored when in 16 bit I/O count by words mode as it is replaced by the bit 15 shifted out from the current address register.			RW

6.1.2.4 Offset 08h: DMACMD—DMA Command Register (LPC I/F—B0:D31:F0)

Table 6-40. Offset 08h: DMACMD—DMA Command Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** Ch.#0 - 3 = 08h **Offset End:** Ch.#4 - 7 = D0h	
Size: 8 bit	**Default:** Undefined			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05	Reserved	Reserved. Must be 0.			
04		DMA Group Arbitration Priority — Each channel group is individually assigned either fixed or rotating arbitration priority. At part reset, each group is initialized in fixed priority. 0 = Fixed priority to the channel group 1 = Rotating priority to the group.			WO
03	Reserved	Reserved. Must be 0.			
02		DMA Channel Group Enable — Both channel groups are enabled following part reset. 0 = Enable the DMA channel group. 1 = Disable. Disabling channel group 4–7 also disables channel group 0–3, which is cascaded through channel 4.			WO
01 :00	Reserved	Reserved. Must be 0.			

6.1.2.5 Offset 08h: DMASTA—DMA Status Register (LPC I/F—B0:D31:F0)

Table 6-41. Offset 08h: DMASTA—DMA Status Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** Ch.#0 - 3 = 08h **Offset End:** Ch.#4 - 7 = D0h	
Size: 8 bit	**Default:** Undefined			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04		Channel Request Status — When a valid DMA request is pending for a channel, the corresponding bit is set to 1. When a DMA request is not pending for a particular channel, the corresponding bit is set to 0. The source of the DREQ may be hardware or a software request. Channel 4 is the cascade channel, so the request status of channel 4 is a logical OR of the request status for channels 0 through 3. 4 = Channel 0 5 = Channel 1 (5) 6 = Channel 2 (6) 7 = Channel 3 (7)			RO
03 :00		Channel Terminal Count Status — When a channel reaches terminal count (TC), its status bit is set to 1. If TC has not been reached, the status bit is set to 0. Channel 4 is programmed for cascade, so the TC bit response for channel 4 is irrelevant: 0 = Channel 0 1 = Channel 1 (5) 2 = Channel 2 (6) 3 = Channel 3 (7)			RO

6.1.2.6 Offset 0Ah: DMA_WRSMSK—DMA Write Single Mask Register (LPC I/F—B0:D31:F0)

Table 6-42. Offset 0Ah: DMA_WRSMSK—DMA Write Single Mask Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** Ch.#0 - 3 = 0Ah **Offset End:** Ch.#4 - 7 = D4h	
Size: 8 bit	**Default:** 0000 01xx			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved. Must be 0.			
02		Channel Mask Select: 0 = Enable DREQ for the selected channel. The channel is selected through bits [1:0]. Therefore, only one channel can be masked / unmasked at a time. 1 = Disable DREQ for the selected channel.			WO
01 :00		DMA Channel Select — These bits select the DMA Channel Mode Register to program. 00 = Channel 0 (4) 01 = Channel 1 (5) 10 = Channel 2 (6) 11 = Channel 3 (7)			WO

6.1.2.7 Offset 0Bh: DMACH_MODE—DMA Channel Mode Register (LPC I/F—B0:D31:F0)

Table 6-43. Offset 08h: DMACH_MODE—DMA Channel Mode Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** Ch.#0 - 3 = 0Bh **Offset End:** Ch.#4 - 7 = D6h	
Size: 8 bit	**Default:** 0000 00xx			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06		DMA Transfer Mode — Each DMA channel can be programmed in one of four different modes: 00 = Demand mode 01 = Single mode 10 = Reserved 11 = Cascade mode			WO
05		Address Increment/Decrement Select — This bit controls address increment/decrement during DMA transfers. 0 = Address increment. (default after part reset or Master Clear) 1 = Address decrement.			WO
04		Autoinitialize Enable: 0 = Autoinitialize feature is disabled and DMA transfers terminate on a terminal count. A part reset or Master Clear disables autoinitialization. 1 = DMA restores the Base Address and Count registers to the current registers following a terminal count (TC).			WO
03 :02		DMA Transfer Type — These bits represent the direction of the DMA transfer. When the channel is programmed for cascade mode, (bits[7:6] = 11) the transfer type is irrelevant. 00 = Verify – No I/O or memory strobes generated 01 = Write – Data transferred from the I/O devices to memory 10 = Read – Data transferred from memory to the I/O device 11 = Invalid			WO
01 :00		DMA Channel Select — These bits select the DMA Channel Mode Register that will be written by bits [7:2]. 00 = Channel 0 (4) 01 = Channel 1 (5) 10 = Channel 2 (6) 11 = Channel 3 (7)			WO

6.1.2.8 Offset 0Ch: DMA Clear Byte Pointer Register (LPC I/F—B0:D31:F0)

Table 6-44. Offset 0Ch: DMA Clear Byte Pointer Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** Ch.#0 - 3 = 0Ch **Offset End:** Ch.#4 - 7 = D8h	
Size: 8 bit	**Default:** XXXX XXXX			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Clear Byte Pointer — No specific pattern. Command enabled with a write to the I/O port address. Writing to this register initializes the byte pointer flip/flop to a known state. It clears the internal latch used to address the upper or lower byte of the 16-bit Address and Word Count Registers. The latch is also cleared by part reset and by the Master Clear command. This command precedes the first access to a 16-bit DMA controller register. The first access to a 16-bit register will then access the significant byte, and the second access automatically accesses the most significant byte.			WO

6.1.2.9 Offset 0Dh: DMA Master Clear Register (LPC I/F—B0:D31:F0)

Table 6-45. Offset 0Dh: DMA Master Clear Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** Ch.#0 - 3 = 0Ch **Offset End:** Ch.#4 - 7 = DAh	
Size: 8 bit	**Default:** XXXX XXXX			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Master Clear — No specific pattern. Enabled with a write to the port. This has the same effect as the hardware Reset. The Command, Status, Request, and Byte Pointer flip/flop registers are cleared and the Mask Register is set.			WO

6.1.2.10 Offset 0Eh: DMA_CLMSK—DMA Clear Mask Register (LPC I/F—B0:D31:F0)

Table 6-46. Offset 0Eh: DMA Master Clear Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** Ch.#0 - 3 = 0Eh **Offset End:** Ch.#4 - 7 = DCh	
Size: 8 bit	**Default:** XXXX XXXX			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Clear Mask Register — No specific pattern. Command enabled with a write to the port.			WO

6.1.2.11 Offset 0Fh: DMA_WRMSK—DMA Write All Mask Register (LPC I/F—B0:D31:F0)

Table 6-47. Offset 0Fh: DMA_WRMSK—DMA Write All Mask Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** Ch.#0 - 3 = 0Fh **Offset End:** Ch.#4 - 7 = DEh	
Size: 8 bit	**Default:** 0000 1111			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved. Must be 0.			
03 :00		Channel Mask Bits —This register permits all four channels to be simultaneously enabled/disabled instead of enabling/disabling each channel individually, as is the case with the Mask Register – Write Single Mask Bit. In addition, this register has a read path to allow the status of the channel mask bits to be read. A channel's mask bit is automatically set to 1 when the Current Byte/Word Count Register reaches terminal count (unless the channel is in auto-initialization mode). Setting the bit(s) to a 1 disables the corresponding DREQ(s). Setting the bit(s) to a 0 enables the corresponding DREQ(s). Bits [3:0] are set to 1 upon part reset or Master Clear. When read, bits [3:0] indicate the DMA channel [3:0] ([7:4]) mask status. Bit 0 = Channel 0 (4)1 = Masked, 0 = Not Masked Bit 1 = Channel 1 (5)1 = Masked, 0 = Not Masked Bit 2 = Channel 2 (6)1 = Masked, 0 = Not Masked Bit 3 = Channel 3 (7)1 = Masked, 0 = Not Masked Disabling channel 4 also disables channels 0–3 due to the cascade of channel's 0 – 3 through channel 4.			RW

6.2 Timer I/O Registers (LPC I/F—B0:D31:F0)

6.2.1 Timer I/O Registers

Table 6-48. Timer I/O Registers

Offset Start	**Offset End**	**Register ID - Description**	**Default Value**
43h	43h	"Offset 43h: TCW—Timer Control Word Register (LPC I/F—B0:D31:F0)" on page 278	Bits[7:1]= undefined, Bit 0=0
40h	42h	"Offset 40h: SBYTE_FMT—Interval Timer Status Byte Format Register (LPC I/F—B0:D31:F0)" on page 280	Bits[6:0] = undefined, Bit[7]=0
40h	42h	"Offset 40h: Counter Access Ports Register (LPC I/F—B0:D31:F0)" on page 281	Undefined

6.2.1.1 Offset 43h: TCW—Timer Control Word Register (LPC I/F—B0:D31:F0)

This register is programmed prior to any counter being accessed to specify counter modes. Following part reset, the control words for each register are undefined and each counter output is 0. Each timer must be programmed to bring it into a known state.

Table 6-49. Offset 43h: TCW—Timer Control Word Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 43h **Offset End:** 43h	
Size: 8 bit	**Default:** Bits[7:1] = undefined, Bit[0]=0			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06		Counter Select — The Counter Selection bits select the counter the control word acts upon as shown below. The Read Back Command is selected when bits[7:6] are both 1. 00 = Counter 0 select 01 = Counter 1 select 10 = Counter 2 select 11 = Read Back Command			WO
05 :04		Read/Write Select — These bits are the read/write control bits. The actual counter programming is done through the counter port (40h for counter 0, 41h for counter 1, and 42h for counter 2). 00 = Counter Latch Command 01 = Read/Write Least Significant Byte (LSB) 10 = Read/Write Most Significant Byte (MSB) 11 = Read/Write LSB then MSB			WO
03 :01		Counter Mode Selection — These bits select one of six possible modes of operation for the selected counter. **Bit Value** — **Mode** 000b — Mode 0 Out signal on end of count (=0) 001b — Mode 1 Hardware retriggerable one-shot x10b — Mode 2 Rate generator (divide by n counter) x11b — Mode 3 Square wave output 100b — Mode 4 Software triggered strobe			WO
00		Binary/BCD Countdown Select: 0 = Binary countdown is used. The largest possible binary count is 2^{16} 1 = Binary coded decimal (BCD) count is used. The largest possible BCD count is 10^{4}			WO

There are two special commands that can be issued to the counters through this register, the Read Back Command and the Counter Latch Command. When these commands are chosen, several bits within this register are redefined. These register formats are described below:

RDBK_CMD—Read Back Command (LPC I/F—B0:D31:F0)

The Read Back Command is used to determine the count value, programmed mode, and current states of the OUT pin and Null count flag of the selected counter or counters. Status and/or count may be latched in any or all of the counters by selecting the counter during the register write. The count and status remain latched until read, and further latch commands are ignored until the count is read. Both count and status of the selected counters may be latched simultaneously by setting both bit 5 and bit 4 to 0. If both are latched, the first read operation from that counter returns the latched status. The next one or two reads, depending on whether the counter is programmed for one or two byte counts, returns the latched count. Subsequent reads return an unlatched count.

Bit	Description
07:06	Read Back Command. Must be 11 to select the Read Back Command
05	Latch Count of Selected Counters. 0 = Current count value of the selected counters will be latched 1 = Current count will not be latched
04	Latch Status of Selected Counters. 0 = Status of the selected counters will be latched 1 = Status will not be latched
03	Counter 2 Select. 1 = Counter 2 count and/or status will be latched
02	Counter 1 Select. 1 = Counter 1 count and/or status will be latched
01	Counter 0 Select. 1 = Counter 0 count and/or status will be latched.
00	Reserved. Must be 0.

LTCH_CMD—Counter Latch Command (LPC I/F—B0:D31:F0)

The Counter Latch Command latches the current count value. This command is used to insure that the count read from the counter is accurate. The count value is then read from each counter's count register through the Counter Ports Access Ports Register (40h for counter 0, 41h for counter 1, and 42h for counter 2). The count must be read according to the programmed format, for example, if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other (read, write, or programming operations for other counters may be inserted between the reads). If a counter is latched once and then latched again before the count is read, the second Counter Latch Command is ignored.

Bit	Description
07:06	Counter Selection. These bits select the counter for latching. If “11” is written, then the write is interpreted as a read back command. 00 = Counter 0 01 = Counter 1 10 = Counter 2
05:04	Counter Latch Command. 00 = Selects the Counter Latch Command.
03:00	Reserved. Must be 0.

6.2.1.2 Offset 40h: SBYTE_FMT—Interval Timer Status Byte Format Register (LPC I/F—B0:D31:F0)

Each counter's status byte can be read following a Read Back Command. If latch status is chosen (bit 4=0, Read Back Command) as a read back option for a given counter, the next read from the counter's Counter Access Ports Register (40h for counter 0, 41h for counter 1, and 42h for counter 2) returns the status byte.

Table 6-50. Offset 40h: SBYTE_FMT—Interval Timer Status Byte Format Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 40h **Offset End:** 42h		
Size: 8 bit per counter	**Default:** Bits[6:0] = undefined, Bit[7]=0		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Counter OUT Pin State: 0 = OUT pin of the counter is also a 0 1 = OUT pin of the counter is also a 1			RO
06		Count Register Status — This bit indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the counter mode, but until the count is loaded into the counting element (CE), the count value will be incorrect. 0 = Count has been transferred from CR to CE and is available for reading. 1 = Null Count. Count has not been transferred from CR to CE and is not yet available for reading.			RO
05 :04		Read/Write Selection Status — These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 = Counter Latch Command 01 = Read/Write Least Significant Byte (LSB) 10 = Read/Write Most Significant Byte (MSB) 11 = Read/Write LSB then MSB			RO
03 :01		Mode Selection Status — These bits return the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. 000 = Mode 0 — Out signal on end of count (=0) 001 = Mode 1 — Hardware retriggerable one-shot x10 = Mode 2 — Rate generator (divide by n counter) x11 = Mode 3 — Square wave output 100 = Mode 4 — Software triggered strobe 101 = Mode 5 — Hardware triggered strobe			RO
00		Countdown Type Status — This bit reflects the current countdown type. 0 = Binary countdown 1 = Binary Coded Decimal (BCD) countdown.			RO

6.2.1.3 Offset 40h: Counter Access Ports Register (LPC I/F—B0:D31:F0)

I/O Address:
Counter 0 = 40h
Counter 1 = 41h
Counter 2 = 42h

Table 6-51. Offset 40h: Counter Access Ports Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 40h **Offset End:** 42h	
Size: 8 bit	**Default:** All bits undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Counter Port — Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.			RW

6.3 8259 Interrupt Controller (PIC) Registers

6.3.1 Interrupt Controller I/O MAP

The interrupt controller registers are located at 20h and 21h for the master controller (IRQ 0–7), and at A0h and A1h for the slave controller (IRQ 8–13). These registers have multiple functions, depending upon the data written to them. Table 6-52 shows the register possibilities for each address.

Table 6-52. PIC Registers

Port	Aliases	Register Name	Default Value	Type
20h	24h, 28h, 2Ch, 30h, 34h, 38h, 3Ch	"Offset 20h: Master PIC ICW1—Master Initialization Command Word 1 Register"	Undefined	WO
		"Offset 020h: Master PIC OCW2—Master Operational Control Word 2 Register"	001XXXXXb	WO
		"Offset 020h: Master PIC OCW3—Master Operational Control Word 3 Register"	X01XXX10b	WO
21h	25h, 29h, 2Dh, 31h, 35h, 39h, 3Dh	"Offset 21h: Master PIC ICW2—Master Initialization Command Word 2 Register"	Undefined	WO
		"Offset 21h: Master PIC ICW3—Master Initialization Command Word 3 Register"	Undefined	WO
		"Offset 021h: Master PIC ICW4—Master Initialization Command Word 4 Register"	01h	WO
		"Offset 021h: Master PIC OCW1—Master Operational Control Word 1 (Interrupt Mask) Register"	00h	R/W
A0h	A4h, A8h, ACh, B0h, B4h, B8h, BCh	Slave PIC ICW1 - Slave Initialization Command Word 1	Undefined	WO
		Slave PIC OCW2 - Slave Operational Control Word 2	001XXXXXb	WO
		Slave PIC OCW3 - Slave Operational Control Word 3	X01XXX10b	WO
A1h	A5h, A9h, ADh, B1h, B5h, B9h, BDh	Slave PIC ICW2 - Slave Initialization Command Word 2	Undefined	WO
		Slave PIC ICW3 - Slave Initialization Command Word 3	Undefined	WO
		Slave PIC ICW4 - Slave Initialization Command Word 4	01h	WO
		Slave PIC OCW1 - Slave Operational Control Word 1	00h	R/W
4D0h	–	"Offset 4D0h: Master PIC ELCR1—Master Controller Edge/Level Triggered Register"	00h	R/W
4D1h	–	"Offset 4D1h: Slave PIC ELCR2—Slave Controller Edge/Level Triggered Register"	00h	R/W

Note: See the note addressing active-low interrupt sources in 8259 Interrupt Controllers section (Chapter 4.6).

6.3.1.1 Offset 20h: Master PIC ICW1—Master Initialization Command Word 1 Register

Offset Address: Master Controller – 20h
Slave Controller – A0h
Attribute: WO
Size: 8 bit /controller
Default Value: All bits undefined

A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

1. The Interrupt Mask register is cleared.
2. IRQ7 input is assigned priority 7.
3. The slave mode address is set to 7.
4. Special mask mode is cleared and Status Read is set to IRR.

Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Table 6-53. Offset 20h: Master PIC ICW1—Master Initialization Command Word 1 Register

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 20h **Offset End:** 20h	
Size: 8 bit per controller	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05		ICW/OCW Select — These bits are MCS-85 specific, and not needed. 000 = Should be programmed to "000"			WO
04		ICW/OCW Select — This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.			WO
03	LTIM	Edge/Level Bank Select — Disabled. Replaced by the edge/level triggered control registers (ELCR, B0:D31:F0:4D0h, B0:D31:F0:4D1h).			WO
02		ADI — Ignored for the PCH. Should be programmed to 0.			WO
01	SNGL	Single or Cascade: Must be programmed to a 0 to indicate two controllers operating in cascade mode.			WO
00	IC4	ICW4 Write Required —This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.			WO

6.3.1.2 Offset 21h: Master PIC ICW2—Master Initialization Command Word 2 Register

Offset Address: Master Controller – 21h
Slave Controller – A1h
Attribute: WO
Size: 8 bit /controller
Default Value: All bits undefined

ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits[7:3] is used by the processor to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Table 6-54. Offset 21h: Master PIC ICW2—Master Initialization Command Word 2 Register

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: IA F</td><td colspan="2">BAR: 000h (IO)</td><td>Bus:Device:Function: B0:D31 :F0</td><td>Offset Start: 21h
Offset End: 21h</td></tr>
<tr><td>Size: 8 bit per controller</td><td colspan="2">Default: Undefined</td><td></td><td colspan="2">Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>07 :03</td><td></td><td>Interrupt Vector Base Address — Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.</td><td></td><td></td><td>WO</td></tr>
<tr><td>02 :00</td><td></td><td>Interrupt Request Level — When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code.
<table>
<tr><th>Code</th><th>Master Interrupt</th><th>Slave Interrupt</th></tr>
<tr><td>000b</td><td>IRQ0</td><td>IRQ8</td></tr>
<tr><td>001b</td><td>IRQ1</td><td>IRQ9</td></tr>
<tr><td>010b</td><td>IRQ2</td><td>IRQ10</td></tr>
<tr><td>011b</td><td>IRQ3</td><td>IRQ11</td></tr>
<tr><td>100b</td><td>IRQ4</td><td>IRQ12</td></tr>
<tr><td>101b</td><td>IRQ5</td><td>IRQ13</td></tr>
<tr><td>110b</td><td>IRQ6</td><td>IRQ14</td></tr>
<tr><td>111b</td><td>IRQ7</td><td>IRQ15</td></tr>
</table></td><td></td><td></td><td>WO</td></tr>
</table>

6.3.1.3 Offset 21h: Master PIC ICW3—Master Initialization Command Word 3 Register

Offset Address: 21h
Default Value: All bits undefined
Attribute: WO
Size: 8 bits

Table 6-55. Offset 21h: Master PIC ICW3—Master Initialization Command Word 3 Register

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 21h **Offset End:** 21h	
Size: 8 bit per controller	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03		0 = These bits must be programmed to 0.			O
02		Cascaded Interrupt Controller IRQ Connection — This bit indicates that the slave controller is cascaded on IRQ2. When IRQ8#–IRQ15 is asserted, it goes through the slave controller's priority resolver. The slave controller's INTR output onto IRQ2. IRQ2 then goes through the master controller's priority solver. If it wins, the INTR signal is asserted to the processor, and the returning interrupt acknowledge returns the interrupt vector for the slave controller. 1 = This bit must always be programmed to a 1.			WO
01 :00		0 = These bits must be programmed to 0.			O

6.3.1.4 Offset A1h: Slave PIC ICW3—Slave Initialization Command Word 3 Register

Offset Address: A1h
Default Value: All bits undefined
Attribute: WO
Size: 8 bits

Table 6-56. Offset A1h: Slave PIC ICW3—Slave Initialization Command Word 3 Register

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A1h **Offset End:** A1h	
Size: 8 bit	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03		0 = These bits must be programmed to 0.			O
02 :00		Slave Identification Code — These bits are compared against the slave identification code broadcast by the master controller from the trailing edge of the first internal INTA# pulse to the trailing edge of the second internal INTA# pulse. These bits must be programmed to 02h to match the code broadcast by the master controller. When 02h is broadcast by the master controller during the INTA# sequence, the slave controller assumes responsibility for broadcasting the interrupt vector.			WO

6.3.1.5 Offset 021h: Master PIC ICW4—Master Initialization Command Word 4 Register

Offset Address: Master Controller – 021h
Slave Controller – 0A1h
Attribute: WO
Size: 8 bits
Default Value: 01h

Table 6-57. Offset 021h: Master PIC ICW4—Master Initialization Command Word 4 Register

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** 021h **Offset End:** 021h	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05		0 = These bits must be programmed to 0.			O
04	SFNM	Special Fully Nested Mode: 0 = Should normally be disabled by writing a 0 to this bit. 1 = Special fully nested mode is programmed.			WO
03	BUF	Buffered Mode: 0 = Must be programmed to 0 for the PCH. This is non-buffered mode.			WO
02		Master/Slave in Buffered Mode — Not used. 0 = Should always be programmed to 0.			WO
01	AEOI	Automatic End of Interrupt: 0 = This bit should normally be programmed to 0. This is the normal end of interrupt. 1 = Automatic End of Interrupt mode is programmed.			WO
00		Microprocessor Mode —Must be programmed to 1 to indicate that the controller is operating in an Intel Architecture-based system.			WO

6.3.1.6 Offset 021h: Master PIC OCW1—Master Operational Control Word 1 (Interrupt Mask) Register

Offset Address: Master Controller – 021h
Slave Controller – 0A1h
Attribute: R/W
Size: 8 bits
Default Value: 00h

Table 6-58. Offset 021h: Master PIC OCW1—Master Operational Control Word 1 (Interrupt Mask) Register

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 021h **Offset End:** 021h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Interrupt Request Mask — When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.			RW

6.3.1.7 Offset 020h: Master PIC OCW2—Master Operational Control Word 2 Register

Offset Address: Master Controller – 020h
Slave Controller – 0A0h
Attribute: WO
Size: 8 bits
Default Value: Bit[4:0]=undefined, Bit[7:5]=001

Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Table 6-59. Offset 020h: Master PIC OCW2—Master Operational Control Word 2 Register

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** 020h **Offset End:** 020h	
Size: 8 bit	**Default:** 001XXXXXb			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05	R, SL, EOI	Rotate and EOI Codes —These three bits control the Rotate and End of Interrupt modes and combinations of the two. 000 = Reserved 001 = Non-specific EOI command 010 = No Operation 011 = *Specific EOI Command 100 = Reserved 101 = Rotate on Non-Specific EOI Command 110 = *Set Priority Command 111 = *Rotate on Specific EOI Command *L0 – L2 Are Used			WO
04 :03		OCW2 Select — When selecting OCW2, bits 4:3 = "00"			WO
02 :00	L2, L1, L0	Interrupt Level Select — L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined below, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function; programming L2, L1 and L0 to 0 is sufficient in this case. **Code / Interrupt Level / Code / Interrupt Level** 000b IRQ0/8 000b IRQ4/12 001b IRQ1/9 001b IRQ5/13 010b IRQ2/10 010b IRQ6/14 011b IRQ3/11 011b IRQ7/15			WO

6.3.1.8 Offset 020h: Master PIC OCW3—Master Operational Control Word 3 Register

Offset Address: Master Controller – 020h Attribute:WO
Slave Controller – 0A0h Size: 8 bits
Default Value: Bit[6,0]=0, Bit[7,4:2]=undefined, Bit[5,1]=1

Table 6-60. Offset 020h: Master PIC OCW3—Master Operational Control Word 3 Register

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 020h **Offset End:** 020h	
Size: 8 bit	**Default:** X01XXX10b			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	Reserved	Reserved. Must be 0.			
06	SMM	Special Mask Mode: 1 = The Special Mask Mode can be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/disabling of the other channel's mask bits. Bit 5, the ESMM bit, must be set for this bit to have any meaning.			WO
05	ESMM	Enable Special Mask Mode. 0 = Disable. The SMM bit becomes a "don't care". 1 = Enable the SMM bit to set or reset the Special Mask Mode.			WO
04 :03		OCW3 Select — When selecting OCW3, bits 4:3 = 01			WO
02		Poll Mode Command: 0 = Disable. Poll Command is not issued. 1 = Enable. The next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.			WO
01 :00		Register Read Command — These bits provide control for reading the In-Service Register (ISR) and the Interrupt Request Register (IRR). When bit 1=0, bit 0 will not affect the register read selection. When bit 1=1, bit 0 selects the register status returned following an OCW3 read. If bit 0=0, the IRR will be read. If bit 0=1, the ISR will be read. Following ICW initialization, the default OCW3 port address read will be "read IRR". To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register can be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. 00 = No Action 01 = No Action 10 = Read IRQ Register 11 = Read IS Register			WO

6.3.1.9 Offset 4D0h: Master PIC ELCR1—Master Controller Edge/Level Triggered Register

Offset Address: 4D0h
Attribute: R/W
Default Value: 00h
Size: 8 bits

In edge mode, (bit[x] = 0), the interrupt is recognized by a low to high transition. In level mode (bit[x] = 1), the interrupt is recognized by a high level. The cascade channel, IRQ2, the heart beat timer (IRQ0), and the keyboard controller (IRQ1), cannot be put into level mode

Table 6-61. Offset 4D0h: Master PIC ELCR1—Master Controller Edge/Level Triggered Register

Description:					
View: IA F	**BAR:** 000h (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 4D0h **Offset End:** 4D0h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		IRQ7 ECL: 0 = Edge. 1 = Level.			RW
06		IRQ6 ECL: 0 = Edge. 1 = Level.			RW
05		IRQ5 ECL: 0 = Edge. 1 = Level.			RW
04		IRQ4 ECL: 0 = Edge. 1 = Level.			RW
03		IRQ3 ECL: 0 = Edge. 1 = Level.			RW
02 :00	Reserved	Reserved. Must be 0.			

6.3.1.10 Offset 4D1h: Slave PIC ELCR2—Slave Controller Edge/Level Triggered Register

Offset Address: 4D1h
Default Value: 00h
Attribute: R/W
Size: 8 bits

In edge mode, (bit[x] = 0), the interrupt is recognized by a low to high transition. In level mode (bit[x] = 1), the interrupt is recognized by a high level. The real time clock, IRQ8#, and the floating point error interrupt, IRQ13, cannot be programmed for level mode.

Table 6-62. Offset 4D1h: Slave PIC ELCR2—Slave Controller Edge/Level Triggered Register

Description:					
View: IA F	**BAR:** 000h (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 4D1h **Offset End:** 4D1h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		IRQ7 ECL: 0 = Edge. 1 = Level.			RW
06		IRQ6 ECL: 0 = Edge. 1 = Level.			RW
05		IRQ5 ECL: 0 = Edge. 1 = Level.			RW
04		IRQ4 ECL: 0 = Edge. 1 = Level.			RW
03		IRQ3 ECL: 0 = Edge. 1 = Level.			RW
02 :00	Reserved	Reserved. Must be 0.			

6.4 Advanced Programmable Interrupt Controller (APIC)

6.4.1 APIC Register Map

The APIC is accessed via an indirect addressing scheme. Two registers are visible by software for manipulation of most of the APIC registers. These registers are mapped into memory space. The address bits 19:12 of the address range are programmable through bits 7:0 of OIC register (Chipset Config Registers (RCBA); Offset 31FEh) The registers are shown in Table 6-63.

6.4.2 APIC Direct Registers

Table 6-63. APIC Direct Registers

Memory Address	Register ID - Description	Default Value
FECx y000h	"IND—Index Register" on page 292	00h
FECx y010h	"DAT—Data Register" on page 293	00000000h
FECx y040h	"EOIR—EOI Register" on page 293	N/A
Note: Memory address nibbles: x,y = bits[19:12] are programmable bits[7:0] of the OIC register		

6.4.2.1 IND—Index Register

Memory Address FECx y000h Attribute: R/W
Default Value: 00h Size: 8 bits

The Index Register will select which APIC indirect register to be manipulated by software. The selector values for the indirect registers are listed in Table 6-67. Software will program this register to select the desired APIC internal register

Table 6-64. IND—Index Register

Description:					
View: IA F	**Memory Address** FECx y000h	**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** y000h **Offset End:** y000h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		APIC Index — This is a 8-bit Index to select which APIC indirect register (Table 6-67) to write or read			RW

6.4.2.2 DAT—Data Register

Memory Address FECx y010h Attribute: R/W
Default Value: 00000000h Size: 32 bits

This is a 32-bit register specifying the data to be read or written to the register pointed to by the Index register. This register can only be accessed in DWord quantities.

Table 6-65. DAT—Data Register

Description:					
View: IA F	**Memory Address** FECx y010h		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** y010h **Offset End:** y013h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		APIC Data — This is a 32-bit register for the data to be read or written to the APIC indirect register (Table 6-67) pointed to by the Index register (Memory Address FEC0 0000h, if OIC Register Bits[7:0] = 00h)			RW

6.4.2.3 EOIR—EOI Register

Memory Address FECx y040h Attribute: R/W
Default Value: N/A Size: 32 bits

The EOI register is present to provide a mechanism to maintain the level triggered semantics for level-triggered interrupts issued on the parallel bus.

When a write is issued to this register, the I/O APIC will check the lower 8 bits written to this register, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit (Index Offset 10h, bit 14) for that I/O Redirection Entry will be cleared.

Note: If multiple I/O Redirection entries, for any reason, assign the same vector for more than one interrupt input, each of those entries will have the Remote_IRR bit reset to 0. The interrupt which was prematurely reset will not be lost because if its input remained active when the Remote_IRR bit is cleared, the interrupt will be reissued and serviced at a later time. Only bits 7:0 are used. Bits 31:8 are ignored by the PCH.

- To provide for future expansion, the processor should always write a value of 0 to Bits 31:8.

Table 6-66. EOIR—EOI Register

Description:					
View: IA F	**Memory Address** FECx y040h		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** y040h **Offset End:** y043h	
Size: 32 bit	**Default:** N/A			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved. To provide for future expansion, the processor should always write a value of 0 to Bits 31:08.			
07 :00		Redirection Entry Clear — When a write is issued to this register, the I/O APIC will check this field, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared.			WO

6.4.3 APIC Indirect Registers

Table 6-67 lists the registers that can be accessed within the APIC via the Index Register defined in Table 6-63. When accessing these registers, accesses must be done one dword at a time. For example, software should never access byte 2 from the Data register before accessing bytes 0 and 1. The hardware will not attempt to recover from a bad programming model in this case.

Table 6-67. APIC Indirect Registers (Sheet 1 of 2)

Index	Mnemonic & Register Name	Size	Type
00h	"Offset 00h: ID—Identification Register" on page 295	32 bits	R/W
01h	"Offset 01h: VER—Version Register" on page 296	32 bits	RO
02–0F	Reserved		RO
10h–11h	"Offset 10h: REDIR_TBL0—Redirection Table 0" on page 296	64 bits	R/W, RO
12h–13h	REDIR_TBTL1 - Redirection Table 1	64 bits	R/W, RO
14h–15h	REDIR_TBTL2 - Redirection Table 2	64 bits	R/W, RO
16h–17h	REDIR_TBTL3 - Redirection Table 3	64 bits	R/W, RO
18h–19h	REDIR_TBTL4 - Redirection Table 4	64 bits	R/W, RO
1Ah–1Bh	REDIR_TBTL5 - Redirection Table 5	64 bits	R/W, RO
1Ch–1Dh	REDIR_TBTL6 - Redirection Table 6	64 bits	R/W, RO
1Eh–1Fh	REDIR_TBTL7 - Redirection Table 7	64 bits	R/W, RO
20h–21h	REDIR_TBTL8 - Redirection Table 8	64 bits	R/W, RO
22h–23h	REDIR_TBTL9 - Redirection Table 9	64 bits	R/W, RO
24h–25h	REDIR_TBTL10 - Redirection Table 10	64 bits	R/W, RO
26h–27h	REDIR_TBTL11 - Redirection Table 11	64 bits	R/W, RO
28h–29h	REDIR_TBTL12 - Redirection Table 12	64 bits	R/W, RO
2Ah–2Bh	REDIR_TBTL13 - Redirection Table 13	64 bits	R/W, RO

Table 6-67. APIC Indirect Registers (Sheet 2 of 2)

Index	Mnemonic & Register Name	Size	Type
2Ch–2Dh	REDIR_TBTL14 - Redirection Table 14	64 bits	R/W, RO
2Eh–2Fh	REDIR_TBTL15 - Redirection Table 15	64 bits	R/W, RO
30h–31h	REDIR_TBTL16 - Redirection Table 16	64 bits	R/W, RO
32h–33h	REDIR_TBTL17 - Redirection Table 17	64 bits	R/W, RO
34h–35h	REDIR_TBTL18 - Redirection Table 18	64 bits	R/W, RO
36h–37h	REDIR_TBTL19 - Redirection Table 19	64 bits	R/W, RO
38h–39h	REDIR_TBTL20 - Redirection Table 20	64 bits	R/W, RO
3Ah–3Bh	REDIR_TBTL21 - Redirection Table 21	64 bits	R/W, RO
3Ch–3Dh	REDIR_TBTL22 - Redirection Table 22	64 bits	R/W, RO
3Eh–3Fh	REDIR_TBTL23 - Redirection Table 23	64 bits	R/W, RO
40h–FFh	Reserved		

6.4.3.1 Offset 00h: ID—Identification Register

Index Offset: 00h Attribute: R/W
Default Value: 00000000h Size: 32 bits

The APIC ID serves as a physical name of the APIC. The APIC bus arbitration ID for the APIC is derived from its I/O APIC ID. This register is reset to 0 on power-up reset.

Table 6-68. Offset 00h: ID—Identification Register

Description:					
View: IA I	**Win:Idx:** APIC_WDW:APIC_IDX		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 00h **Offset End:** 00h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
27 :24		APIC ID — This field contains the APIC Identification. Software must program this value before using the APIC.			RW
23 :16	Reserved	Reserved			
15		Scratchpad Bit.			RW
14 :00	Reserved	Reserved			

6.4.3.2 Offset 01h: VER—Version Register

Index Offset: 01h
Attribute: RO, RWO
Default Value: 00170020h
Size: 32 bits

Each I/O APIC contains a hardwired Version Register that identifies different implementation of APIC and their versions. The maximum redirection entry information also is in this register, to let software know how many interrupt are supported by this APIC.

Table 6-69. Offset 01h: VER—Version Register

Description:					
View: IA I	**Win:Idx:** APIC_WDW:APIC_IDX		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 01h **Offset End:** 01h	
Size: 32 bit	**Default:** 00170020h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :16	MRE	Maximum Redirection Entries — This is the entry number (0 being the lowest entry) of the highest entry in the redirection table. It is equal to the number of interrupt input pins minus one and is in the range 0 through 239. In the PCH this field is hardwired to 17h to indicate 24 interrupts. BIOS must write to this field after PLTRST# to lockdown the value. this allows BIOS to utilize some of the entries for its own purpose and thus advertising fewer IOxAPIC Redirection Entries to the OS.			RWO
15	PRQ	Pin Assertion Register Supported — Indicate that the IOxAPIC does not implement the Pin Assertion Register.			RO
14 :08	Reserved	Reserved			
07 :00	VS	Version — This is a version number that identifies the implementation version.			RO

6.4.3.3 Offset 10h: REDIR_TBL0—Redirection Table 0

Index Offset: 10h–11h (vector 0)
12h-13h (vector 1)

3E–3Fh (vector 23)
Attribute:R/W, RO
Default Value: Bit 16 = 1. All other bits undefined
Size: 64 bits each, (accessed as two 32 bit quantities)

The Redirection Table has a dedicated entry for each interrupt input pin. The information in the Redirection Table is used to translate the interrupt manifestation on the corresponding interrupt pin into an APIC message.

The APIC will respond to an edge triggered interrupt as long as the interrupt is held until after the acknowledge cycle has begun. Once the interrupt is detected, a delivery status bit internally to the I/O APIC is set. The state machine will step ahead and wait for an acknowledgment from the APIC unit that the interrupt message was sent. Only then will the I/O APIC be able to recognize a new edge on that interrupt pin. That new

edge will only result in a new invocation of the handler if its acceptance by the destination APIC causes the Interrupt Request Register bit to go from 0 to 1. (In other words, if the interrupt was not already pending at the destination.)

Table 6-70. Offset 10h: REDIR_TBL0—Redirection Table 0 (Sheet 1 of 2)

Description:					
View: IA I	**Win:Idx:** APIC_WDW:APIC_IDX		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 10h **Offset End:** 11h	
Size: 64 bits	**Default:** Bit 16 = 1. All other bits undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :56	DEST	Destination — If bit 11 of this entry is 0 (Physical), then bits 59:56 specifies an APIC ID. In this case, bits 63:59 should be programmed by software to 0. If bit 11 of this entry is 1 (Logical), then bits 63:56 specify the logical destination address of a set of processors.			RW
55 :48	EDID	Extended Destination ID — These bits are sent to a local APIC only when in Processor System Bus mode. They become bits 11:4 of the address.			RO
47 :17	Reserved	Reserved			
16	MASK	Mask: 0 = Not masked: An edge or level on this interrupt pin results in the delivery of the interrupt to the destination. 1 = Masked: Interrupts are not delivered nor held pending. Setting this bit after the interrupt is accepted by a local APIC has no effect on that interrupt. This behavior is identical to the device withdrawing the interrupt before it is posted to the processor. It is software's responsibility to deal with the case where the mask bit is set after the interrupt message has been accepted by a local APIC unit but before the interrupt is dispensed to the processor.			RW
15	TRIGMOD	Trigger Mode — This field indicates the type of signal on the interrupt pin that triggers an interrupt. 0 = Edge triggered. 1 = Level triggered.			RW
14		Remote IRR — This bit is used for level triggered interrupts; its meaning is undefined for edge triggered interrupts. 0 = Reset when an EOI message is received from a local APIC. 1 = Set when Local APIC/s accept the level interrupt sent by the I/O APIC.			RO
13	INTPOL	Interrupt Input Pin Polarity — This bit specifies the polarity of each interrupt signal connected to the interrupt pins. 0 =Active high. 1 = Active low.			RW
12	DELIVS	Delivery Status — This field contains the current status of the delivery of this interrupt. Writes to this bit have no effect. 0 = Idle. No activity for this interrupt. 1 = Pending. Interrupt has been injected, but delivery is not complete.			RO

Table 6-70. Offset 10h: REDIR_TBL0—Redirection Table 0 (Sheet 2 of 2)

Description:					
View: IA I	**Win:Idx:** APIC_WDW:APIC_IDX		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 10h **Offset End:** 11h	
Size: 64 bits	**Default:** Bit 16 = 1. All other bits undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	DESTMOD	Destination Mode — This field determines the interpretation of the Destination field. 0 = Physical. Destination APIC ID is identified by bits 59:56. 1 = Logical. Destinations are identified by matching bit 63:56 with the Logical Destination in the Destination Format Register and Logical Destination Register in each Local APIC			RW
10 :08	DELMOD	Delivery Mode — This field specifies how the APICs listed in the destination field should act upon reception of this signal. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. These encodings are listed in the note below.			RW
07 :00	INTVEC	Interrupt Vector — This field contains the interrupt vector for this interrupt. Values range between 10h and FEh.			RW

Note: **Bits[10:8] - Delivery Mode Encoding**

000 = Fixed. Deliver the signal on the INTR signal of all processor cores listed in the destination. Trigger Mode can be edge or level.

001 = Lowest Priority. Deliver the signal on the INTR signal of the processor core that is executing at the lowest priority among all the processors listed in the specified destination. Trigger Mode can be edge or level.

010 = SMI (System Management Interrupt). Requires the interrupt to be programmed as edge triggered. The vector information is ignored but must be programmed to all 0s for future compatibility: **not supported**

011 = Reserved

100 = NMI. Deliver the signal on the NMI signal of all processor cores listed in the destination. Vector information is ignored. NMI is treated as an edge triggered interrupt even if it is programmed as level triggered. For proper operation this redirection table entry must be programmed to edge triggered. The NMI delivery mode does not set the RIRR bit. If the redirection table is incorrectly set to level, the loop count will continue counting through the redirection table addresses. Once the count for the NMI pin is reached again, the interrupt will be sent again: **not supported**

101 = INIT. Deliver the signal to all processor cores listed in the destination by asserting the INIT signal. All addressed local APICs will assume their INIT state. INIT is always treated as an edge triggered interrupt even if programmed as level triggered. For proper operation this redirection table entry must be programmed to edge triggered. The INIT delivery mode does not set the RIRR bit. If the redirection table is incorrectly set to level, the loop count will continue counting through the redirection table addresses. Once the count for the INIT pin is reached again, the interrupt will be sent again: **not supported**

110 = Reserved

111 = ExtINT. Deliver the signal to the INTR signal of all processor cores listed in the destination as an interrupt that originated in an externally connected 8259A compatible interrupt controller. The INTA cycle that corresponds to this ExtINT delivery will be routed to the external controller that is expected to supply the vector. Requires the interrupt to be programmed as edge triggered.

6.5 Real Time Clock Registers

6.5.1 I/O Register Address Map

The RTC internal registers and RAM are organized as two banks of 128 bytes each, called the standard and extended banks. The first 14 bytes of the standard bank contain the RTC time and date information along with four registers, A–D, that are used for configuration of the RTC. The extended bank contains a full 128 bytes of battery backed SRAM, and will be accessible even when the RTC module is disabled (via the RTC configuration register). Registers A–D do not physically exist in the RAM.

All data movement between the host processor and the real-time clock is done through registers mapped to the standard I/O space. The register map appears in Table 6-71.

Table 6-71. RTC I/O Registers

I/O Locations	If U128E bit = 0	Function
70h and 74h	Also alias to 72h and 76h	Real-Time Clock (Standard RAM) Index Register
71h and 75h	Also alias to 73h and 77h	Real-Time Clock (Standard RAM) Target Register
72h and 76h		Extended RAM Index Register (if enabled)
73h and 77h		Extended RAM Target Register (if enabled)

Notes:

1. I/O locations 70h and 71h are the standard legacy location for the real-time clock. The map for this bank is shown in Table 6-72. Locations 72h and 73h are for accessing the extended RAM. The extended RAM bank is also accessed using an indexed scheme. I/O address 72h is used as the address pointer and I/O address 73h is used as the data register. Index addresses above 127h are not valid. If the extended RAM is not needed, it may be disabled.
2. Software must preserve the value of bit 7 at I/O addresses 70h and 74h. When writing to this address, software must first read the value, and then write the same value for bit 7 during the sequential address write. Port 70h is not directly readable. The only way to read this register is through Alt Access mode. Although RTC Index bits 6:0 are readable from port 74h, bit 7 will always return 0. If the NMI# enable is not changed during normal operation, software can alternatively read this bit once and then retain the value for all subsequent writes to port 70h.

6.5.2 Indexed Registers

The RTC contains two sets of indexed registers that are accessed using the two separate Index and Target registers (70/71h or 72/73h), as shown in Table 6-72.

Table 6-72. RTC (Standard) RAM Bank

Index	Name
00h	Seconds
01h	Seconds Alarm
02h	Minutes
03h	Minutes Alarm
04h	Hours
05h	Hours Alarm
06h	Day of Week
07h	Day of Month
08h	Month
09h	Year
0Ah	Register A
0Bh	Register B
0Ch	Register C
0Dh	Register D
0Eh–7Fh	114 Bytes of User RAM

Table 6-73. Real Time Clock Registers

Offset Start	Offset End	Register ID - Description	Default Value
0Ah	0Ah	"Offset 0Ah: RTC_REGA—Register A" on page 301	Undefined
0Bh	0Bh	"Offset 0Bh: RTC_REGB—Register B (General Configuration)" on page 302	X0X00XXX
0Ch	0Ch	"Offset OCh: RTC_REGC—Register C (Flag Register)" on page 303	00X00000
0Dh	0Dh	"Offset ODh: D—Register D (Flag Register)" on page 304	10XXXXXX

6.5.2.1 Offset 0Ah: RTC_REGA—Register A

This register is used for general configuration of the RTC functions. None of the bits are affected by RSMRST# or any other PCH reset signal.

Table 6-74. Offset 0Ah: RTC_REGA—Register A

Description:					
View: IA F	**BAR:** RTC Standard RAM Bank	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** Undefined			**Power Well:** RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	UIP	Update In Progress — This bit may be monitored as a status flag. 0 =The update cycle will not start for at least 488 µs. The time, calendar, and alarm information in RAM is always available when the UIP bit is 0. 1 = The update is soon to occur or is in progress.			RW
06 :04	DV[2:0]	Division Chain Select — These three bits control the divider chain for the oscillator, and are not affected by RSMRST# or any other reset signal. 010 = Normal Operation 11X = Divider Reset 101 = Bypass 15 stages (test mode only) 100 = Bypass 10 stages (test mode only) 011 = Bypass 5 stages (test mode only) 001 = Invalid 000 = Invalid			RW
03 :00	RS[3:0]	Rate Select — Selects one of 13 taps of the 15 stage divider chain. The selected tap can generate a periodic interrupt if the PIE bit is set in Register B. Otherwise this tap will set the PF flag of Register C. If the periodic interrupt is not to be used, these bits should all be set to 0. RS3 corresponds to bit 3. 0000 = Interrupt never toggles 0001 = 3.90625 ms 0010 = 7.8125 ms 0011 = 122.070 µs 0100 = 244.141 µs 0101 = 488.281 µs 0110 = 976.5625 µs 0111 = 1.953125 ms 1000 = 3.90625 ms 1001 = 7.8125 ms 1010 = 15.625 ms 1011 = 31.25 ms 1100 = 62.5 ms 1101 = 125 ms 1110 = 250 ms 1111= 500 ms			RW

6.5.2.2 Offset 0Bh: RTC_REGB—Register B (General Configuration)

Table 6-75. Offset 0Bh: RTC_REGB—Register B (General Configuration)

Description:					
View: IA F	**BAR:** RTC Standard RAM Bank		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** X0X00XXX			**Power Well:** RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	SET	Update Cycle Inhibit — Enables/Inhibits the update cycles. This bit is not affected by RSMRST# nor any other reset signal. 0 = Update cycle occurs normally once each second. 1 = A current update cycle will abort and subsequent update cycles will not occur until SET is returned to 0. When set is one, the BIOS may initialize time and calendar bytes safely. This bit should be set then cleared early in BIOS POST after each powerup directly after coin-cell battery insertion.			RW
06	PIE	Periodic Interrupt Enable — This bit is cleared by RSMRST#, but not on any other reset. 0 = Disable. 1 = Enable. Allows an interrupt to occur with a time base set with the RS bits of register A.			RW
05	AIE	Alarm Interrupt Enable — This bit is cleared by RTCRST#, but not on any other reset. 0 = Disable. 1 = Enable. Allows an interrupt to occur when the AF is set by an alarm match from the update cycle. An alarm can occur once a second, one an hour, once a day, or one a month.			RW
04	UIE	Update-Ended Interrupt Enable — This bit is cleared by RSMRST#, but not on any other reset. 0 = Disable. 1 = Enable. Allows an interrupt to occur when the update cycle ends.			RW
03	SQWE	Square Wave Enable — This bit serves no function in the PCH. It is left in this register bank to provide compatibility with the Motorola 146818B. The PCH has no SQW pin. This bit is cleared by RSMRST#, but not on any other reset.			RW
02	DM	Data Mode — This bit specifies either binary or BCD data representation. This bit is not affected by RSMRST# nor any other reset signal. 0 = BCD 1 = Binary			RW
01	HOURFORM	Hour Format —This bit indicates the hour byte format. This bit is not affected by RSMRST# nor any other reset signal. 0 = Twelve-hour mode. In twelve-hour mode, the seventh bit represents AM as 0 and PM as one. 1 = Twenty-four hour mode.			RW
00	DSLSWS	Daylight Savings Legacy Software Support — Daylight savings functionality is no longer supported. This bit is used to maintain legacy software support and has no associated functionality. If BUC.DSO bit is set, the DSLSWS bit continues to be R/W.			RW

6.5.2.3 Offset OCh: RTC_REGC—Register C (Flag Register)

Writes to Register C have no effect.

Table 6-76. Offset OCh: RTC_REGC—Register C (Flag Register)

Description:					
View: IA F	**BAR:** RTC Standard RAM Bank		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00X00000			**Power Well:** RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	IRQF	Interrupt Request Flag — IRQF = (PF * PIE) + (AF * AIE) + (UF *UFE). This bit also causes the RTC Interrupt to be asserted. This bit is cleared upon RSMRST# or a read of Register C.			RO
06	PF	Periodic Interrupt Flag — This bit is cleared upon RSMRST# or a read of Register C. 0 = If no taps are specified via the RS bits in Register A, this flag will not be set. 1 = Periodic interrupt Flag will be 1 when the tap specified by the RS bits of register A is 1.			RO
05	AF	Alarm Flag: 0 = This bit is cleared upon RTCRST# or a read of Register C. 1 = Alarm Flag will be set after all Alarm values match the current time.			RO
04	UF	Update-Ended Flag: 0 = The bit is cleared upon RSMRST# or a read of Register C. 1 = Set immediately following an update cycle for each second.			RO
03 :00	Reserved	Reserved. Will always report 0.			

6.5.2.4 Offset ODh: D—Register D (Flag Register)

Table 6-77. Offset ODh: D—Register D (Flag Register)

Description:					
View: IA F	**BAR:** RTC Standard RAM Bank		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 10XXXXXX			**Power Well:** RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	VRT	Valid RAM and Time Bit: 0 = This bit should always be written as a 0 for write cycle, however it will return a 1 for read cycles. 1 = This bit is hardwired to 1 in the RTC power well.			RW
06	Reserved	Reserved. This bit always returns a 0 and should be set to 0 for write cycles.			
05 :00		Date Alarm — These bits store the date of month alarm value. If set to 000000b, then a don't care state is assumed. The host must configure the date alarm for these bits to do anything, yet they can be written at any time. If the date alarm is not enabled, these bits will return 0s to mimic the functionality of the Motorola 146818B. These bits are not affected by any reset assertion.			RW

6.6 Processor Interface Registers (LPC I/F—B0:D31:F0)

6.6.1 Processor Interface PCI Registers Address Map

The following table provides the register address map for the processor interface registers.

Table 6-78. Processor Interface PCI Register Address Map (LPC I/F—B0:D31:F0)

Offset Start	Offset End	Register ID - Description	Default Value
61h	61h	"Offset 61h: NMI_SC—NMI Status and Control Register (LPC I/F—B0:D31:F0)" on page 306	00h
70h	70h	"Offset 70h: NMI_EN—NMI Enable (and Real Time Clock Index) Register (LPC I/F—B0:D31:F0)" on page 307	80h
92h	92h	"Offset 92h: PORT92—Fast A20 and Init Register (LPC I/F—B0:D31:F0)" on page 307	00h
F0h	F0h	"Offset F0h: COPROC_ERR—Coprocessor Error Register (LPC I/F—B0:D31:F0)" on page 308	00h
CF9h	CF9h	"Offset CF9h: RST_CNT—Reset Control Register (LPC I/F—B0:D31:F0)" on page 308	00h
B2h	B2h	"Offset B2h: APM_CNT—Advanced Power Management Control Port Register" on page 321	00h
B3h	B3h	"Offset B3h: APM_STS—Advanced Power Management Status Port Register" on page 321	00h

6.6.1.1 Offset 61h: NMI_SC—NMI Status and Control Register (LPC I/F—B0:D31:F0)

Table 6-79. Offset 61h: NMI_SC—NMI Status and Control Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 61h **Offset End:** 61h		
Size: 8 bit	**Default:** 00h		**Power Well:** Core		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	SERR#_NMI_STS	SERR# NMI Source Status: 1 = Bit is set if a PCI agent detected a system error and pulses the PCI SERR# line and if bit 2 (PCI_SERR_EN) is cleared. This interrupt source is enabled by setting bit 2 to 0. To reset the interrupt, set bit 2 to 1 and then set it to 0. When writing to port 61h, this bit must be 0. Note: This bit is set by any of the PCH internal sources of SERR; this includes SERR assertions forwarded from the secondary PCI bus, errors on a PCI Express* port, or other internal functions that generate SERR#.			RO
06	IOCHK_NMI_STS	IOCHK# NMI Source Status: 1 = Bit is set if an LPC agent (via SERIRQ) asserted IOCHK# and if bit 3 (IOCHK_NMI_EN) is cleared. This interrupt source is enabled by setting bit 3 to 0. To reset the interrupt, set bit 3 to 1 and then set it to 0. When writing to port 61h, this bit must be a 0.			RO
05	TMR2_OUT_STS	Timer Counter 2 OUT Status: This bit reflects the current state of the 8254 counter 2 output. Counter 2 must be programmed following any PCI reset for this bit to have a determinate value. When writing to port 61h, this bit must be a 0.			RO
04	REF_TOGGLE	Refresh Cycle Toggle: This signal toggles from either 0 to 1 or 1 to 0 at a rate that is equivalent to when refresh cycles would occur. When writing to port 61h, this bit must be a 0.			RO
03	IOCHK_NMI_EN	IOCHK# NMI Enable: 0 = Enabled. 1 = Disabled and cleared.			RW
02	PCI_SERR_EN	PCI SERR# Enable: 0 = SERR# NMIs are enabled. 1 = SERR# NMIs are disabled and cleared.			RW
01	SPKR_DAT_EN	Speaker Data Enable: 0 = SPKR output is a 0. 1 = SPKR output is equivalent to the Counter 2 OUT signal value.			RW
00	TIM_CNT2_EN	Timer Counter 2 Enable: 0 = Disable 1 = Enable.			RW

6.6.1.2 Offset 70h: NMI_EN—NMI Enable (and Real Time Clock Index) Register (LPC I/F—B0:D31:F0)

Note: The RTC Index field is write-only for normal operation. This field can only be read in Alt-Access Mode. This register is aliased to Port 74h (documented in Table 6-71), and all bits are readable at that address.

Table 6-80. Offset 70h: NMI_EN—NMI Enable (and Real Time Clock Index) Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 70h **Offset End:** 70h	
Size: 8 bit	**Default:** 80h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	NMI_EN	NMI Enable (special): 0 = Enable NMI sources. 1 = Disable All NMI sources.			RW
06 :00	RTC_INDX	Real Time Clock Index Address (special): This data goes to the RTC to select which register or CMOS RAM address is being accessed.			RW

6.6.1.3 Offset 92h: PORT92—Fast A20 and Init Register (LPC I/F—B0:D31:F0)

Table 6-81. Offset 92h: PORT92—Fast A20 and Init Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 92h **Offset End:** 92h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :02	Reserved	Reserved			
01	ALT_A20_GATE	Alternate A20 Gate — This bit is Or'd with the A20GATE input signal to generate A20M# to the processor. 0 = A20M# signal can potentially go active. 1 = This bit is set when INIT# goes active.			RW
00	INIT_NOW	When this bit transitions from a 0 to a 1, the PCH will force INIT# active for 16 PCI clocks.			RW

6.6.1.4 Offset F0h: COPROC_ERR—Coprocessor Error Register (LPC I/F—B0:D31:F0)

Table 6-82. Offset F0h: COPROC_ERR—Coprocessor Error Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** F0h **Offset End:** F0h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	COPROC_ERR	Coprocessor Error — Any value written to this register will cause IGNNE# to go active, if FERR# had generated an internal IRQ13. For FERR# to generate an internal IRQ13, the COPROC_ERR_EN bit must be 1.			WO

6.6.1.5 Offset CF9h: RST_CNT—Reset Control Register (LPC I/F—B0:D31:F0)

Table 6-83. Offset CF9h: RST_CNT—Reset Control Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** CF9h **Offset End:** CF9h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved			
03	FULL_RST	Full Reset — This bit is used to determine the states of SLP_S3#, SLP_S4#, and SLP_S5# after a CF9 hard reset (SYS_RST =1 and RST_CPU is set to 1), after PWROK going low (with RSMRST# high), or after two TCO timeouts. 0 = PCH will keep SLP_S3#, SLP_S4# and SLP_S5# high. 1 = PCH will drive SLP_S3#, SLP_S4# and SLP_S5# low for 3 – 5 seconds. Note: When this bit is set, it also causes the full power cycle (SLP_S3/4/5# assertion) in response to SYS_RESET#, PWROK#, and Watchdog timer reset sources.			RW

Table 6-83. Offset CF9h: RST_CNT—Reset Control Register (LPC I/F—B0:D31:F0)

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** CF9h **Offset End:** CF9h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	RST_CPU	Reset CPU — When this bit transitions from a 0 to a 1, it initiates a hard or soft reset, as determined by the SYS_RST bit (bit 1 of this register).			RW
01	SYS_RST	System Reset — This bit is used to determine a hard or soft reset to the processor. 0 = When RST_CPU bit goes from 0 to 1, the PCH performs a soft reset by activating INIT# for 16 PCI clocks. 1 = When RST_CPU bit goes from 0 to 1, the PCH performs a hard reset by activating PLTRST# and SUS_STAT# active for a minimum of about 1 milliseconds. In this case, SLP_S3#, SLP_S4# and SLP_S5# state (assertion or de-assertion) depends on FULL_RST bit setting. The PCH main power well is reset when this bit is 1. It also resets the resume well bits (except for those noted throughout the EDS).			RW
00	Reserved	Reserved			

6.7 Power Management Registers (PM—B0:D31:F0)

The power management registers are distributed within the PCI Bus 0:Device 31: Function 0 space, as well as a separate I/O range. Each register is described below. Unless otherwise indicate, bits are in the main (core) power well.

Bits not explicitly defined in each register are assumed to be reserved. When writing to a reserved bit, the value should always be 0. Software should not attempt to use the value read from a reserved bit, as it may not be consistently 1 or 0.

6.7.1 Power Management PCI Configuration Registers (PM—B0:D31:F0)

Table 6-84 shows a small part of the configuration space for PCI Bus 0:Device 31: Function 0. It includes only those registers dedicated for power management. Some of the registers are only used for legacy power management schemes.

Table 6-84. Power Management PCI Register Address Map (PM—B0:D31:F0)

Offset Start	Offset End	Register ID - Description	Default Value
A0h	A0h	"Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0)" on page 311	0000h
A2h	A2h	"Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0)" on page 313	00h
A4h	A4h	"Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0)" on page 315	00h
A6h	A6h	"Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register" on page 318	00h
A9h	A9h	"Offset A9h: Chipset Initialization Register 4 (PM—B0:D31:F0)" on page 318	01h
ABh	ABh	"Offset ABh: BM_BREAK_EN Register (PM—B0:D31:F0)" on page 319	01h
ACh	ACh	"Offset ACh: PMIR—Power Management Initialization Register (PM—B0:D31:F0)" on page 319	00000000h
B8h	BBh	"Offset B8h: GPIO_ROUT—GPIO Routing Control Register (PM—B0:D31:F0)" on page 320	00000000h

6.7.1.1 Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0)

Table 6-85. Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A0h **Offset End:** A0h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved			
10	BIOS_PCI_EXP_EN	This bit acts as a global enable for the SCI associated with the PCI Express* ports. 0 = The various PCI Express* ports and Processor cannot cause the PCI_EXP_STS bit to go active. 1 = The various PCI Express* ports and Processor can cause the PCI_EXP_STS bit to go active.			RW
09	PWRBTN_LVL	This bit indicates the current state of the PWRBTN# signal. 0 = Low. 1 = High.			RO
08 :05	Reserved	Reserved			
04	SMI_LOCK	When this bit is set, writes to the GLB_SMI_EN bit (PMBASE + 30h, bit 0) will have no effect. Once the SMI_LOCK bit is set, writes of 0 to SMI_LOCK bit will have no effect (for example, once set, this bit can only be cleared by PLTRST#).			RWO
03	Reserved	Reserved			
03 (server only)	PSEUDO_CLKRUN_EN	0 = Disable. 1 = Enable internal CLKRUN# logic to allow DMI PLL shutdown. This bit has no impact on state of external CLKRUN# pin. Notes: 1. PSEUDO_CLKRUN_EN bit does not result in STP_PCI# assertion to actually stop the external PCICLK. 2. This bit should be set mutually exclusive with the CLKRUN_EN bit. Setting PSEUDO_CLKRUN_EN in a mobile sku could result in unspecified behavior.			RW

Table 6-85. Offset A0h: GEN_PMCON_1—General PM Configuration 1 Register (PM—B0:D31:F0) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A0h **Offset End:** A0h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	CLKRUN_EN	PCI CLKRUN# Enable: 0 = Disable. Drives the CLKRUN# signal low. 1 = Enable CLKRUN# logic to control the system PCI clock via the CLKRUN# and STP_PCI# signals. Notes: 1. When the SLP_EN# bit is set, the PCH drives the CLKRUN# signal low regardless of the state of the CLKRUN_EN bit. This ensures that the PCI and LPC clocks continue running during a transition to a sleep state. 2. This bit should be set mutually exclusive with the PSEUDO_CLKRUN_EN bit. Setting CLKRUN_EN in a non-mobile sku could result in unspecified behavior.			RW
02	Reserved	Reserved			
01 :00	PER_SMI_SEL	Periodic SMI# Rate Select: Set by software to control the rate at which periodic SMI# is generated. 00 = 64 seconds 01 = 32 seconds 10 = 16 seconds 11 = 8 seconds			RW

6.7.1.2 Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0)

Table 6-86. Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** A2h **Offset End:** A2h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		DRAM Initialization Bit — This bit does not effect hardware functionality in any way. BIOS is expected to set this bit prior to starting the DRAM initialization sequence and to clear this bit after completing the DRAM initialization sequence. BIOS can detect that a DRAM initialization sequence was interrupted by a reset by reading this bit during the boot sequence. - If the bit is 1, then the DRAM initialization was interrupted. - This bit is reset by the assertion of the RSMRST# pin.			RW
06	Reserved	Reserved			
05	MEM_SR	Memory Placed in Self-Refresh: - If the bit is 1, DRAM should have remained powered and held in Self-Refresh through the last power state transition (for example, the last time the system left S0). - This bit is reset by the assertion of the RSMRST# pin.			RO
04	SRS	System Reset Status — Software clears this bit by writing a 1 to it. 0 = SYS_RESET# button Not pressed. 1 = PCH sets this bit when the SYS_RESET# button is pressed. BIOS is expected to read this bit and clear it, if it is set. Notes: 1. This bit is also reset by RSMRST# and CF9h resets. 2. The SYS_RESET# is implemented in the Main power well. This pin must be properly isolated and masked to prevent incorrectly setting this Suspend well status bit.			RWC
03	CTS	CPU Thermal Trip Status: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when PLTRST# is inactive and THRMTRIP# goes active while the system is in an S0 or S1 state. Notes: 1. This bit is also reset by RSMRST#, and CF9h resets. It is not reset by the shutdown and reboot associated with the CPUTHRMTRIP# event. 2. The CF9h reset in the description refers to CF9h type core well reset which includes SYS_RST#, PWROK/SYS_PWROK low, SMBus hard reset, TCO Timeout. This type of reset will clear CTS bit.			RWC

Table 6-86. Offset A2h: GEN_PMCON_2—General PM Configuration 2 Register (PM—B0:D31:F0) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A2h **Offset End:** A2h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02		Minimum SLP_S4# Assertion Width Violation Status: 0 = Software clears this bit by writing a 1 to it. 1 = Hardware sets this bit when the SLP_S4# assertion width is less than the time programmed in the SLP_S4# Minimum Assertion Width field (B0:D31:F0:Offset A4h:bits 5:4). The PCH begins the timer when SLP_S4# is asserted during S4/S5 entry, or when the RSMRST# input is deasserted during G3 exit. NOTE: This bit is functional regardless of the value in the SLP_S4# Assertion Stretch Enable (B0:D31:F0:Offset A4h:bit 3). NOTE: This bit is reset by the assertion of the RSMRST# pin, but can be set in some cases before the default value is readable.			RWC
01	Reserved	Reserved			
00	PWROK_FLR	PWROK Failure: 0 = Software clears this bit by writing a 1 to it, or when the system goes into a G3 state. 1 = This bit will be set any time PWROK goes low, when the system was in S0, or S1 state.			RWC

6.7.1.3 Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0)

Table 6-87. Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0) (Sheet 1 of 3)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D31 :F0</td><td>Offset Start: A4h
Offset End: A4h</td></tr>
<tr><td colspan="2">Size: 16 bit</td><td colspan="2">Default: 00h</td><td></td><td>Power Well: RTC, SUS</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>15</td><td>PME_B0_S5_DIS</td><td>PME B0 S5 Disable —When set to '1', this bit blocks wake events from PME_B0_STS in S5, regardless of the state of PME_B0_EN. When cleared (default), wake events from PME_B0_STS are allowed in S5 if PME_B0_EN = '1'.
Wakes from power states other than S5 are not affected by this policy bit.
The net effect of setting PME_B0_S5_DIS = '1' is described by the truth table below:
Y = Wake; N = Don't wake; B0 = PME_B0_EN
<table><tr><th>B0/OV</th><th>S1/S3/S4</th><th>S5</th></tr><tr><td>00</td><td>N</td><td>N</td></tr><tr><td>01</td><td>N</td><td>N</td></tr><tr><td>11</td><td>Y (all PME B0 sources)</td><td>N</td></tr><tr><td>01</td><td>Y (all PME B0 sources)</td><td>N</td></tr></table>This bit is cleared by the RTCRST# pin.</td><td></td><td></td><td>RW</td></tr>
<tr><td>14</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>13</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>12</td><td></td><td>Disable SLP_S4# Stretching after G3:
0 = Enables stretching on SLP_S4# in conjunction with SLP_S4# Assertion Stretch Enable (bit 3) and the Minimum Assertion Width (bits 5:4)
1 = Disables stretching on SLP_S4# regardless of the state of the SLP_S4# Assertion Stretch Enable (bit 3).
This bit is cleared by the RTCRST# pin.
Note: This field is RO when the SLP_Sx# Stretching Policy Lock- Down bit is set.</td><td></td><td></td><td>RW</td></tr>
<tr><td>11 :10</td><td></td><td>SLP_S3# Minimum Assertion Width — This 2-bit value indicates the minimum assertion width of the SLP_S3# signal to guarantee that the Main power supplies have been fully power-cycled.
Valid Settings are:
00: 60-100 us
01: 1-1.2 ms
10: 50-50.2 ms
11: 2-2.0002 s
This bit is cleared by the RSMRST# pin.
This field is RO when the SLP_Sx# Stretching Policy Lock-Down bit is set.</td><td></td><td></td><td>RW</td></tr>
</table>

Table 6-87. Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A4h **Offset End:** A4h	
Size: 16 bit	**Default:** 00h			**Power Well:** RTC, SUS	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	GEN_RST_STS	General Reset Status — This bit is set by hardware whenever PLTRST# asserts for any reason other than going into a software-entered sleep state (via PM1CNT.SLP_EN write) or a suspend well power failure (RSMRST# pin assertion). BIOS is expected to consult and then write a '1' to clear this bit during the boot flow before determining what action to take based on PM1_STS.WAK_STS = 1. If GEN_RST_STS = '1', the cold reset boot path should be followed rather than the resume path, regardless of the setting of WAK_STS. This bit is cleared by the RSMRST# pin.			RWC
08	Reserved	Reserved			
07 :06	SWSMI_RATE_SEL	This field indicates when the SWSMI timer will time out. Valid values are: 00 = 1.5 ms ± 0.6 ms 01 = 16 ms ± 4 ms 10 = 32 ms ± 4 ms 11 = 64 ms ± 4 ms These bits are not cleared by any type of reset except RTCRST#.			RW
05 :04		SLP_S4# Minimum Assertion Width — This field indicates the minimum assertion width of the SLP_S4# signal to ensure that the DRAMs have been safely power-cycled. Valid values are: 11 = 1 second 10 = 2 seconds 01 = 3 seconds 00 = 4 seconds This value is used in two ways: 1. If the SLP_S4# assertion width is ever shorter than this time, a status bit is set for BIOS to read when S0 is entered. 2. If enabled by bit 3 in this register, the hardware will prevent the SLP_S4# signal from deasserting within this minimum time period after asserting. RTCRST# forces this field to the conservative default state (00b). This field is RO when the SLP_S4# Stretching Policy Lock-Down bit is set.			RW/RO
03		SLP_S4# Assertion Stretch Enable: 0 = The SLP_S4# minimum assertion time is 1 to 2 RTCCLK. 1 = The SLP_S4# signal minimally assert for the time specified in bits 5:4 of this register. This bit is cleared by RTCRST#. This bit is RO when the SLP_S4# Stretching Policy Lock-Down bit is set.			RW/RO

Table 6-87. Offset A4h: GEN_PMCON_3—General PM Configuration 3 Register (PM—B0:D31:F0) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A4h **Offset End:** A4h	
Size: 16 bit	**Default:** 00h			**Power Well:** RTC, SUS	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	RTC_PWR_STS	RTC Power Status — This bit is set when RTCRST# indicates a weak or missing battery. The bit is not cleared by any type of reset. The bit will remain set until the software clears it by writing a 0 back to this bit position.			RW
01	PWR_FLR	Power Failure — This bit is in the RTC well, and is not cleared by any type of reset except RTCRST#. 0 = Indicates that the trickle current has not failed since the last time the bit was cleared. Software clears this bit by writing a 1 to it. 1 = Indicates that the trickle current (from the main battery or trickle supply) was removed or failed. Clearing CMOS in a PCH-based platform can be done by using a jumper on RTCRST# or GPI, or using SAFEMODE strap. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low.			RWC
00	AFTERG3_EN	This bit determines what state to go to when power is re-applied after a power failure (G3 state). This bit is in the RTC well and is only cleared by writes of 06h or 0Eh to CF9h (when the CF9h global reset bit is clear), receiving hard reset command with or without power cycle from SMBus or RTCRST#. 0 = System will return to S0 state (boot) after power is re-applied. 1 = System will return to the S5 state (except if it was in S4, in which case it will return to S4). In the S5 state, the only enabled wake event is the Power Button or any enabled wake event that was preserved through the power failure. This bit is set any time a Power Button Override occurs (for example, the power button is pressed for at least 4 consecutive seconds), due to the corresponding bit in the SMBus unconditional power down message, due to an internal thermal sensor catastrophic condition and the assertion of THRMTRIP#.			RW

Note: RSMRST# is sampled using the RTC clock. Therefore, low times that are less than one RTC clock period may not be detected by the PCH.

6.7.1.4 Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register

c

Table 6-88. Offset A6h: GEN_PMCON_LOCK- General Power Management Configuration Lock Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A6h **Offset End:** A6h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved			
02	SLP_S4#	Stretching Policy Lock-Down — When set to 1, this bit locks down the SLP_S4# Minimum Assertion Width, the SLP_S4# Assertion Stretch Enable, the Disable SLP_S4# Stretching after G3 and SLP_S4# Assertion Stretch Enable bits in the GEN_PMCON_3 register, making them read-only. This bit becomes locked when a value of 1b is written to it. Writes of 0 to this bit are always ignored. This bit is cleared by platform reset.			RWLO
01	ACPI_BASE_LOCK	When set to 1, this bit locks down the ACPI Base Address Register (ABASE) at offset 40h. The Base Address Field becomes read-only. This bit becomes locked when a value of 1b is written to it. Writes of 0 to this bit are always ignored. Once locked by writing 1, the only way to clear this bit is to perform a platform reset.			RWLO
00	Reserved	Reserved			

6.7.1.5 Offset A9h: Chipset Initialization Register 4 (PM—B0:D31:F0)

Table 6-89. Offset A9h: Chipset Initialization Register 4 (PM—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** A9h **Offset End:** A9h	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		CIR4 Field 1 — BIOS must program this field to 45h.			RW

6.7.1.6 Offset ABh: BM_BREAK_EN Register (PM—B0:D31:F0)

Table 6-90. Offset ABh: BM_BREAK_EN Register (PM—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** ABh **Offset End:** ABh	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	STORAGE_BREAK_EN	0 = Serial ATA traffic will not act as a break event. 1 = Serial ATA traffic acts as a break event, Serial ATA master activity will cause BM_STS to be set and will cause a break from C3/C4.			RW
06	PCIE_BREAK_EN	0 = PCI Express* traffic will not act as a break event. 1 = PCI Express* traffic acts as a break event, PCI Express* master activity will cause BM_STS to be set and will cause a break from C3/C4.			RW
05	Reserved	Reserved			
04 :03	Reserved	Reserved			
02	EHCI_BREAK_EN	0 = EHCI traffic will not act as a break event. 1 = EHCI traffic acts as a break event, EHCI master activity will cause BM_STS to be set and will cause a break from C3/C4.			RW
01		CIR4 Field 1 — BIOS must program this field to 45h.			RW
00		CIR4 Field 1 — BIOS must program this field to 45h.			RW

6.7.1.7 Offset ACh: PMIR—Power Management Initialization Register (PM—B0:D31:F0)

Table 6-91. Offset ACh: PMIR—Power Management Initialization Register (PM—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** ACh **Offset End:** ACh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30		PMIR Field 1— BIOS must program these bits to 11b.			RW
29 :21	Reserved	Reserved			
20	CF9GR	CF9h Global Reset: When set, a CF9h write of 6h or Eh will cause a Global reset of both the Host and Intel® ME partitions. If this bit is cleared, a CF9h write of 6h or Eh will only reset the host partition. This bit field is not reset by a CF9h reset			RW
19 :00		PMIR Field 0 — BIOS must program these bits to 00300h.			RW

6.7.1.8 Offset B8h: GPIO_ROUT—GPIO Routing Control Register (PM—B0:D31:F0)

Table 6-92. Offset B8h: GPIO_ROUT—GPIO Routing Control Register (PM—B0:D31:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30		GPIO15 Route — See bits 1:0 for description.			RW
		Same pattern for GPIO14 through GPIO3.			
05 :04		GPIO2 Route — See bits 1:0 for description.			RW
03 :02		GPIO1 Route — See bits 1:0 for description.			RW
01 :00		GPIO0 Route — GPIO can be routed to cause an NMI, SMI# or SCI when the GPIO[n]_STS bit is set. If the GPIO0 is not set to an input, this field has no effect. If the system is in an S1–S5 state and if the GPE0_EN bit is also set, then the GPIO can cause a Wake event, even if the GPIO is NOT routed to cause an NMI, SMI# or SCI. 00 = No effect. 01 = SMI# (if corresponding ALT_GPI_SMI_EN bit is also set) 10 = SCI (if corresponding GPE0_EN bit is also set) 11 = NMI (If corresponding GPI_NMI_EN is also set)			RW

Note: GPIOs that are not implemented will not have the corresponding bits implemented in this register.

6.7.2 APM I/O Decode

Table 6-93 shows the I/O registers associated with APM support. This register space is enabled in the PCI Bus 0:Device 31: Function 0 space (APMDEC_EN), and cannot be moved (fixed I/O location).

Table 6-93. APM Register Map

Offset Start	**Offset End**	**Register ID - Description**	**Default Value**
B2h	B2h	"Offset B2h: APM_CNT—Advanced Power Management Control Port Register" on page 321	00h
B3h	B3h	"Offset B3h: APM_STS—Advanced Power Management Status Port Register" on page 321	00h

6.7.2.1 Offset B2h: APM_CNT—Advanced Power Management Control Port Register

Table 6-94. Offset B2h: APM_CNT—Advanced Power Management Control Port Register

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** B2h **Offset End:** B2h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Used to pass an APM command between the OS and the SMI handler. Writes to this port not only store data in the APMC register, but also generates an SMI# when the APMC_EN bit is set.			RW

6.7.2.2 Offset B3h: APM_STS—Advanced Power Management Status Port Register

Table 6-95. Offset B3h: APM_STS—Advanced Power Management Status Port Register

Description:					
View: IA F	**BAR:** 0000h (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** B3h **Offset End:** B3h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Used to pass data between the OS and the SMI handler. Basically, this is a scratchpad register and is not affected by any other register or function (other than a PCI reset).			RW

6.7.3 Power Management I/O Registers

The following table shows the registers associated with ACPI and Legacy power management support. These registers are enabled in the PCI Bus 0: Device 31: Function 0 space (PM_IO_EN), and can be moved to any I/O location (128-byte aligned). The registers are defined to support the ACPI 3.0a specification, and use the same bit names.

Note: All reserved bits and registers will always return 0 when read, and will have no effect when written.

Table 6-96. ACPI and Legacy I/O Register Map

Offset Start	Offset End	Register ID - Description	Default Value
00h	B3h	"Offset B3h: APM_STS—Advanced Power Management Status Port Register" on page 321	00h 3642537: Offset PMBASE+00h : PM1_STSóPower Management 1 Status Register (Sheet 1 of 3) 0000h
02h	02h	"Offset PMBASE + 02h: PM1_EN—Power Management 1 Enable Register" on page 326	0000h
04h	04h	"Offset PMBASE + 04h: PM1_CNT—Power Management 1 Control" on page 327	00000000h
08h	08h	"Offset PMBASE + 08h: PM1_TMR—Power Management 1 Timer Register" on page 328	xx000000h
20h	20h	"Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register" on page 329	0000000000000000h
28h	28h	"Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register" on page 331	0000000000000000h
30h	30h	"Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register" on page 333	00000002h
34h	34h	"Offset PMBASE + 34h: SMI_STS—SMI Status Register" on page 335	00000000h
38h	38h	"Offset PMBASE +38h: ALT_GP_SMI_EN—Alternate GPI SMI Enable Register" on page 338	0000h
3Ah	3Ah	"Offset PMBASE +3Ah: ALT_GP_SMI_STS—Alternate GPI SMI Status Register" on page 338	0000h
3Ch	3Ch	"Offset PMBASE +3Ch: UPRWC—USB Per-Port Registers Write Control" on page 339	0000h
42h	42h	"Offset PMBASE +42h: GPE_CNTL—General Purpose Control Register" on page 340	00h
44h	44h	"Offset PMBASE +44h: DEVACT_STS—Device Activity Status Register" on page 341	0000h
50h	50h	"Offset PMBASE +50h: PM2_CNT—Power Management 2 Control" on page 342	00h

6.7.3.1 Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register

If bit 10 or 8 in this register is set, and the corresponding _EN bit is set in the PM1_EN register, then the PCH will generate a Wake Event. Once back in an S0 state (or if already in an S0 state when the event occurs), the PCH will also generate an SCI if the SCI_EN bit is set, or an SMI# if the SCI_EN bit is not set.

Note: Bit 5 does not cause an SMI# or a wake event. Bit 0 does not cause a wake event but can cause an SMI# or SCI..

Table 6-97. Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 00h **Offset End:** 00h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	Bits 0–7: Core, Bits 8–15: Resume, except Bit 11 in RTC
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	WAK_STS	Wake Status — This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#. 0 = Software clears this bit by writing a 1 to it. 1 = Set by hardware when the system is in one of the sleep states (via the SLP_EN bit) and an enabled wake event occurs. Upon setting this bit, the PCH will transition the system to the ON state. If the AFTERG3_EN bit is not set and a power failure (such as removed batteries) occurs without the SLP_EN bit set, the system will return to an S0 state when power returns, and the WAK_STS bit will not be set. If the AFTERG3_EN bit is set and a power failure occurs without the SLP_EN bit having been set, the system will go into an S5 state when power returns, and a subsequent wake event will cause the WAK_STS bit to be set. Any subsequent wake event would have to be caused by either a Power Button press, or an enabled wake event that was preserved through the power failure (enable bit in the RTC well).			RWC
14	PCIEXPWAK_STS	PCI Express* Wake Status: 0 = Software clears this bit by writing a 1 to it. If the WAKE# pin is still active during the write or the PME message received indication has not been cleared in the root port, then the bit will remain active (for example, all inputs to this bit are level-sensitive). 1 = This bit is set by hardware to indicate that the system woke due to a PCI Express* wakeup event. This wakeup event can be caused by the PCI Express* WAKE# pin being active or receipt of a PCI Express* PME message at a root port. This bit is set only when one of these events causes the system to transition from a non-S0 system power state to the S0 system power state. This bit is set independent of the state of the PCIEXP_WAKE_DIS bit. Note: This bit does not itself cause a wake event or prevent entry to a sleeping state. Thus if the bit is 1 and the system is put into a sleeping state, the system will not automatically wake.			RWC
13 :12	Reserved	Reserved			

Table 6-97. Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 00h **Offset End:** 00h		
Size: 16 bit	**Default:** 0000h		**Power Well:** Bits 0–7: Core, Bits 8–15: Resume, except Bit 11 in RTC		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	PWRBTNOR_STS	Power Button Override Status: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set any time a Power Button Override occurs (for example, the power button is pressed for at least 4 consecutive seconds), due to the corresponding bit in the SMBus slave message, Intel® ME Initiated Power Button Override, Intel® ME Initiated Host Reset with Power down or due to an internal thermal sensor catastrophic condition. The power button override causes an unconditional transition to the S5 state, as well as sets the AFTERG3_EN bit. The BIOS or SCI handler clears this bit by writing a 1 to it. This bit is not affected by hard resets via CF9h writes, and is not reset by RSMRST#. Thus, this bit is preserved through power failures. If this bit is still asserted when the global SCI_EN is set then an SCI will be generated.			RWC
10	RTC_STS	RTC Status — This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#: 0 = Software clears this bit by writing a 1 to it. 1 = Set by hardware when the RTC generates an alarm (assertion of the IRQ8# signal). Additionally if the RTC_EN bit (PMBASE + 02h, bit 10) is set, the setting of the RTC_STS bit will generate a wake event.			RWC
09	ME_STS	Management Engine Status — This bit is set when the Intel® Management Engine generates a Non-Maskable wake event, and is not affected by any other enable bit. When this bit is set, the Host Power Management logic wakes to S0. This bit is only set by hardware and can only be reset by writing a one to this bit position. This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#.			RWC
08	PWRBTN_STS	Power Button Status — This bit is not affected by hard resets caused by a CF9 write: 0 = If the PWRBTN# signal is held low for more than 4 seconds, the hardware clears the PWRBTN_STS bit, sets the PWRBTNOR_STS bit, and the system transitions to the S5 state with only PWRBTN# enabled as a wake event. 1 = This bit can be cleared by software by writing a one to the bit position. This bit is set by hardware when the PWRBTN# signal is asserted Low, independent of any other enable bit. In the S0 state, while PWRBTN_EN and PWRBTN_STS are both set, an SCI (or SMI# if SCI_EN is not set) will be generated. In any sleeping state S1–S5, while PWRBTN_EN (PMBASE + 02h, bit 8) and PWRBTN_STS are both set, a wake event is generated. If the PWRBTN_STS bit is cleared by software while the PWRBTN# signal is sell asserted, this will not cause the PWRBN_STS bit to be set. The PWRBTN# signal must go inactive and active again to set the PWRBTN_STS bit.			RWC

Table 6-97. Offset PMBASE+00h: PM1_STS—Power Management 1 Status Register (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 00h **Offset End:** 00h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	Bits 0–7: Core, Bits 8–15: Resume, except Bit 11 in RTC
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	Reserved	Reserved			
05	GBL _STS	Global Status: 0 = The SCI handler should then clear this bit by writing a 1 to the bit location. 1 = Set when an SCI is generated due to BIOS wanting the attention of the SCI handler. BIOS has a corresponding bit, BIOS_RLS, which will cause an SCI and set this bit.			RWC
04	BM_STS	Bus Master Status — This bit will not cause a wake event, SCI or SMI#. 0 = Software clears this bit by writing a 1 to it. Set by the PCH when a PCH-visible bus master requests access to memory or the BM_BUSY# signal is active.			RWC
03 :01	Reserved	Reserved			
00	TO_STS	Timer Overflow Status: 0 = The SCI or SMI# handler clears this bit by writing a 1 to the bit location. 1 = This bit gets set any time bit 22 of the 24-bit timer goes high (bits are numbered from 0 to 23). This will occur every 2.3435 seconds. When the TMROF_EN bit (PMBASE + 02h, bit 0) is set, then the setting of the TMROF_STS bit will additionally generate an SCI or SMI# (depending on the SCI_EN).			RWC

6.7.3.2 Offset PMBASE + 02h: PM1_EN—Power Management 1 Enable Register

Table 6-98. Offset PMBASE + 02h: PM1_EN—Power Management 1 Enable Register

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 02h **Offset End:** 02h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	Bits 0–7: Core, Bits 8–9, 11–15: Resume, Bit 10: RTC
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			
14	PCIEXPWAK_DIS	PCI Express* Wake Disable — Modification of this bit has no impact on the value of the PCIEXP_WAKE_STS bit. 0 = Inputs to the PCIEXP_WAKE_STS bit in the PM1 Status register enabled to wake the system. 1 = Inputs to the PCIEXP_WAKE_STS bit in the PM1 Status register disabled from waking the system.			RW
13 :11	Reserved	Reserved			
10	RTC_EN	RTC Event Enable — This bit is in the RTC well to allow an RTC event to wake after a power failure. This bit is not cleared by any reset other than RTCRST# or a Power Button Override event. 0 = No SCI (or SMI#) or wake event is generated then RTC_STS (PMBASE + 00h, bit 10) goes active. 1 = An SCI (or SMI#) or wake event will occur when this bit is set and the RTC_STS bit goes active.			RW
09	Reserved	Reserved			
08	PWRBTN_EN	Power Button Enable — This bit is used to enable the setting of the PWRBTN_STS bit to generate a power management event (SMI#, SCI). PWRBTN_EN has no effect on the PWRBTN_STS bit (PMBASE + 00h, bit 8) being set by the assertion of the power button. The Power Button is always enabled as a Wake event. 0 = Disable. 1 = Enable.			RW
07 :06	Reserved	Reserved			
05	GBL_EN	Global Enable — When both the GBL_EN and the GBL_STS bit (PMBASE + 00h, bit 5) are set, an SCI is raised. 0 = Disable. 1 = Enable SCI on GBL_STS going active.			RW
04 :01	Reserved	Reserved			
00	TMROF_EN	Timer Overflow Interrupt Enable — Works in conjunction with the SCI_EN bit (PMBASE + 04h, bit 0) as described below: **TMROF_EN / SCI_EN / Effect when TMROF_STS is set** 0 / X / No SMI# or SCI 1 / 0 / SMI# 1 / 1 / SCI			RW

6.7.3.3 Offset PMBASE + 04h: PM1_CNT—Power Management 1 Control

Table 6-99. Offset PMBASE + 04h: PM1_CNT—Power Management 1 Control

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 04h **Offset End:** 04h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Bits 0–7: Core, Bits 8–12: RTC, Bits 13–15: Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13	SLP_EN	Sleep Enable — Setting this bit causes the system to sequence into the Sleep state defined by the SLP_TYP field.			WO
12 :10	SLP_TYP	Sleep Type — This 3-bit field defines the type of Sleep the system should enter when the SLP_EN bit is set to 1. These bits are only reset by RTCRST#. **Code** / **Master Interrupt** 000b: ON: Typically maps to S0 state. 001b: Puts CPU in S1 state. 010b: Reserved 011b: Reserved 100b: Reserved 101b: Suspend-To-RAM. Assert SLP_S3#: Typically maps to S3 state. 110b: Suspend-To-Disk. Assert SLP_S3#, and SLP_S4#: Typically maps to S4 state. 111b: Soft Off. Assert SLP_S3#, SLP_S4#, and SLP_S5#: Typically maps to S5 state.			RW
09 :03	Reserved	Reserved			
02	GBL_RLS	Global Release: 0 = This bit always reads as 0. 1 = ACPI software writes a 1 to this bit to raise an event to the BIOS. BIOS software has a corresponding enable and status bits to control its ability to receive ACPI events.			WO
01	BM_RLD	Bus Master Reload — This bit is treated as a scratchpad bit. This bit is reset to 0 by PLTRST# 0 = Bus master requests will not cause a break from the C3 state. 1 = Enables Bus Master requests (internal or external) to cause a break from the C3 state. If software fails to set this bit before going to C3 state, the PCH will still return to a snoopable state from C3 or C4 states due to bus master activity.			RW
00	SCI_EN	SCI Enable — Selects the SCI interrupt or the SMI# interrupt for various events including the bits in the PM1_STS register (bit 10, 8, 0), and bits in GPE0_STS. 0 = These events will generate an SMI#. 1 = These events will generate an SCI.			RW

6.7.3.4 Offset PMBASE + 08h: PM1_TMR—Power Management 1 Timer Register

Table 6-100. Offset PMBASE + 08h: PM1_TMR—Power Management 1 Timer Register

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 08h **Offset End:** 08h	
Size: 32 bit	**Default:** xx000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :00	TMR_VAL	Timer Value — Returns the running count of the PM timer. This counter runs off a 3.579545 MHz clock (14.31818 MHz divided by 4). It is reset to 0 during a PCI reset, and then continues counting as long as the system is in the S0 state. After an S1 state, the counter will not be reset (it will continue counting from the last value in S0 state. Anytime bit 22 of the timer goes HIGH to LOW (bits referenced from 0 to 23), the TMROF_STS bit (PMBASE + 00h, bit 0) is set. The High-to-Low transition will occur every 2.3435 seconds. If the TMROF_EN bit (PMBASE + 02h, bit 0) is set, an SCI interrupt is also generated.			RO

6.7.3.5 Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register

This register is symmetrical to the General Purpose Event 0 Enable Register. Unless indicated otherwise below, if the corresponding _EN bit is set, then when the _STS bit is set, the PCH will generate a Wake Event. Once back in an S0 state (or if already in an S0 state when the event occurs), the PCH will also generate an SCI if the SCI_EN bit is set, or an SMI# if the SCI_EN bit (PMBASE + 04h, bit 0) is not set. Bits 31:16 are reset by a CF9h write; bits 63:32 and 15:00 are not. All are reset by RSMRST#.

Table 6-101. Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 20h **Offset End:** 20h	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :36	Reserved	Reserved			
35	GPIO27_STS	0 = Disable. 1 = Set by hardware and can be reset by writing a one to this bit position or a resume well reset. This bit is set at the level specified in GP27IO_POL. GPIO27 is always monitored as an input for the purpose of setting this bit, regardless of the actual GPIO configuration.			RWC
34 :32	Reserved	Reserved			
31 :16	GPIOn_STS	0 = Software clears this bit by writing a 1 to it. 1 = These bits are set any time the corresponding GPIO is set up as an input and the corresponding GPIO signal is high (or low if the corresponding GP_INV bit is set). If the corresponding enable bit is set in the GPE0_EN register, then when the GPIO[n]_STS bit is set: - If the system is in an S1–S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), a SCI will be caused depending on the GPIO_ROUT bits (B0:D31:F0:B8h, bits 31:30) for the corresponding GPI. Mapping is as follows: bit 31 corresponds to GPIO[15]... and bit 16 corresponds to GPIO[0].			RWC
15 :14	Reserved	Reserved			
13	PME_B0_STS	This bit will be set to 1 by the PCH when any internal device with PCI Power Management capabilities on bus 0 asserts the equivalent of the PME# signal. Additionally, if the PME_B0_EN bit is set, and the system is in an S0 state, then the setting of the PME_B0_STS bit will generate an SCI (or SMI# if SCI_EN is not set). If the PME_B0_STS bit is set, and the system is in an S1–S4 state (or S5 state due to SLP_TYP and SLP_EN), then the setting of the PME_B0_STS bit will generate a wake event, and an SCI (or SMI# if SCI_EN is not set) will be generated. If the system is in an S5 state due to power button override, then the PME_B0_STS bit will not cause a wake event or SCI. The default for this bit is 0. Writing a 1 to this bit position clears this bit. Note: HD audio wake events are reported in this bit. *Intel*® Management Engine "maskable" wake events are also reported in this bit.			RWC
12	Reserved	Reserved			

Table 6-101. Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 20h **Offset End:** 20h	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	PME_STS	0 = Software clears this bit by writing a 1 to it. 1 = Set by hardware when the PME# signal goes active. Additionally, if the PME_EN bit is set, and the system is in an S0 state, then the setting of the PME_STS bit will generate an SCI or SMI# (if SCI_EN is not set). If the PME_EN bit is set, and the system is in an S1–S4 state (or S5 state due to setting SLP_TYP and SLP_EN), then the setting of the PME_STS bit will generate a wake event, and an SCI will be generated. If the system is in an S5 state due to power button override or a power failure, then PME_STS will not cause a wake event or SCI.			RWC
10	Reserved	Reserved			
09	PCI_EXP_STS	0 = Software clears this bit by writing a 1 to it. 1 = Set by hardware to indicate that: - The PME event message was received on one or more of the PCI Express* ports - An Assert PMEGPE message received from the Processor via DMI Notes: 1 - The PCI WAKE# pin has no impact on this bit. 2 - If the PCI_EXP_STS bit went active due to an Assert PMEGPE message, then a Deassert PMEGPE message must be received prior to the software write in order for the bit to be cleared. 3 - If the bit is not cleared and the corresponding PCI_EXP_EN bit is set, the level-triggered SCI will remain active. 4 - A race condition exists where the PCI Express* device sends another PME message because the PCI Express* device was not serviced within the time when it must resend the message. This may result in a spurious interrupt, and this is comprehended and approved by the *PCI Express* Specification, Revision 1.0a*. The window for this race condition is approximately 95-105 milliseconds.			RWC
08	RI_STS	0 = Software clears this bit by writing a 1 to it. 1 = Set by hardware when the RI# input signal goes active.			RWC
07	SMB_WAK_STS	SMBus Wake Status — The SMBus controller can independently cause an SMI# or SCI, so this bit does not need to do so (unlike the other bits in this register). Software clears this bit by writing a 1 to it. 0 = Wake event Not caused by the PCH's SMBus logic. 1 = Set by hardware to indicate that the wake event was caused by the PCH's SMBus logic.This bit will be set by the WAKE/SMI# command type, even if the system is already awake. The SMI handler should then clear this bit. Notes: 1. The SMBus controller will independently cause an SMI# so this bit does not need to do so (unlike the other bits in this register). 2. This bit is set by the SMBus slave command 01h (Wake/SMI#) even when the system is in the S0 state. Therefore, to avoid an instant wake on subsequent transitions to sleep states, software must clear this bit after each reception of the Wake/SMI# command or just prior to entering the sleep state. 3. The SMBALERT_STS bit (D31:F3:I/O Offset 00h:Bit 5) should be cleared by software before the SMB_WAK_STS bit is cleared.			RWC

Table 6-101. Offset PMBASE + 20h: GPE0_STS—General Purpose Event 0 Status Register (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 20h **Offset End:** 20h	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	TCOSCI_STS	Software clears this bit by writing a 1 to it. 0 = TOC logic or thermal sensor logic did Not cause SCI. 1 = Set by hardware when the TCO logic or thermal sensor logic causes an SCI.			RWC
05 :03	Reserved	Reserved			
02	SWGPE_STS	The SWGPE_CTRL bit (bit 1 of GPE_CTRL reg) acts as a level input to this bit.			RWC
01	HOT_PLUG_STS	0 = This bit is cleared by writing a 1 to this bit position. 1 = When a PCI Express* Hot-Plug event occurs. This will cause an SCI if the HOT_PLUG_EN bit is set in the GEP0_EN register.			RWC
00	Reserved	Reserved			

6.7.3.6 Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register

This register is symmetrical to the General Purpose Event 0 Status Register. All the bits in this register should be cleared to 0 based on a Power Button Override or processor Thermal Trip event. The resume well bits are all cleared by RSMRST#. The RTC well bits are cleared by RTCRST#.

Table 6-102. Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 28h **Offset End:** 28h	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Bits 0–7, 9, 12, 14–63 Resume, Bits 8, 10–11, 13 RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :36	Reserved	Reserved			
35	GPIO27_EN	0 = Disable. 1 = Enable the setting of the GPIO27_STS bit to generate a wake event/SCI/SMI#.			RW
34 :32	Reserved	Reserved			
31 :16	GPIn_EN	These bits enable the corresponding GPI[n]_STS bits being set to cause a SCI, and/or wake event. These bits are cleared by RSMRST#. Mapping is as follows: bit 31 corresponds to GPIO15... and bit 16 corresponds to GPIO0.			RW

Table 6-102. Offset PMBASE + 28h: GPE0_EN—General Purpose Event 0 Enables Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 28h **Offset End:** 28h	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Bits 0–7, 9, 12, 14–63 Resume, Bits 8, 10–11, 13 RTC	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13	PME_B0_EN	0 = Disable 1 = Enables the setting of the PME_B0_STS bit to generate a wake event and/or an SCI or SMI#. PME_B0_STS can be a wake event from the S1–S4 states, or from S5 (if entered via SLP_TYP and SLP_EN) or power failure, but not Power Button Override. This bit defaults to 0. ***Note:*** It is only cleared by Software or RTCRST#. It is not cleared by CF9h writes.			RW
12	Reserved	Reserved			
11	PME_EN	0 = Disable. 1 = Enables the setting of the PME_STS to generate a wake event and/or an SCI. PME# can be a wake event from the S1 – S4 state or from S5 (if entered via SLP_EN, but not power button override).			RW
10	Reserved	Reserved			
09	PCI_EXP_EN	0 = Disable SCI generation upon PCI_EXP_STS bit being set. 1 = Enables PCH to cause an SCI when PCI_EXP_STS bit is set. This is used to allow the PCI Express* ports, including the link to the Processor, to cause an SCI due to wake/PME events.			RW
08	RI_EN	The value of this bit will be maintained through a G3 state and is not affected by a hard reset caused by a CF9h write. 0 = Disable. 1 = Enables the setting of the RI_STS to generate a wake event.			RW
07	Reserved	Reserved			
06	TCOSCI_EN	0 = Disable. 1 = Enables the setting of the TCOSCI_STS to generate an SCI.			RW
05 :03	Reserved	Reserved			
02	SWGPE_EN	This bit allows software to control the assertion of SWGPE_STS bit. This bit This bit, when set to 1, enables the SW GPE function. If SWGPE_CTRL is written to a 1, hardware will set SWGPE_STS (acts as a level input) If SWGPE_STS, SWGPE_EN, and SCI_EN are all 1's, an SCI will be generated If SWGPE_STS = 1, SWGPE_EN = 1, SCI_EN = 0, and GBL_SMI_EN = 1 then an SMI# will be generated.			RW
01	HOT_PLUG_EN	0 = Disables SCI generation upon the HOT_PLUG_STS bit being set. 1 = Enables the PCH to cause an SCI when the HOT_PLUG_STS bit is set. This is used to allow the PCI Express* ports to cause an SCI due to hot-plug events.			RW
00	Reserved	Reserved			

6.7.3.7 Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register

Note: This register is symmetrical to the SMI status register.

Table 6-103. Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** 30h **Offset End:** 30h	
Size: 32 bit	**Default:** 00000002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
:27	GPIO_UNLOCK_SMI_EN	Setting this bit will cause the PCH to generate an SMI# when the GPIO_UNLOCK_SMI_STS bit is set in the SMI_STS register. Once written to '1', this bit can only be cleared by PLTRST#.			RWO
26 :19	Reserved	Reserved			
18	INTEL_USB2_EN	0 = Disable 1 = Enables Intel-Specific USB2 SMI logic to cause SMI#.			RW
17	LEGACY_USB2_EN	0 = Disable 1 = Enables legacy USB2 logic to cause SMI#.			RW
16 :15	Reserved	Reserved			
14	PERIODIC_EN	0 = Disable. 1 = Enables the PCH to generate an SMI# when the PERIODIC_STS bit (PMBASE + 34h, bit 14) is set in the SMI_STS register (PMBASE + 34h).			RW
13	TCO_EN	0 = Disables TCO logic generating an SMI#. If the NMI2SMI_EN bit is set, SMIs that are caused by re-routed NMIs will not be gated by the TCO_EN bit. Even if the TCO_EN bit is 0, NMIs will still be routed to cause SMIs. 1 = Enables the TCO logic to generate SMI#. This bit cannot be written once the TCO_LOCK bit is set.			RW
12	Reserved	Reserved			
11	MCSMI_EN	MCSMI_EN Microcontroller SMI Enable: 0 = Disable. 1 = Enables PCH to trap accesses to the microcontroller range (62h or 66h) and generate an SMI#. "Trapped" cycles will be claimed by the PCH on PCI, but not forwarded to LPC.			RW
10 :08	Reserved	Reserved			
07	BIOS_RLS	BIOS Release: 0 = This bit will always return 0 on reads. Writes of 0 to this bit have no effect. 1 = Enables the generation of an SCI interrupt for ACPI software when a one is written to this bit position by BIOS software. Note: GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place.			WO

Table 6-103. Offset PMBASE + 30h: SMI_EN—SMI Control and Enable Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 30h **Offset End:** 30h	
Size: 32 bit	**Default:** 00000002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	SWSMI_TMR_EN	Software SMI# Timer Enable: 0 = Disable. Clearing the SWSMI_TMR_EN bit before the timer expires will reset the timer and the SMI# will not be generated. Starts Software SMI# Timer. When the SWSMI timer expires (the timeout period depends upon the SWSMI_RATE_SEL bit setting), SWSMI_TMR_STS is set and an SMI# is generated. SWSMI_TMR_EN stays set until cleared by software.			RW
05	APMC_EN	0 = Disable. Writes to the APM_CNT register will not cause an SMI#. Enables writes to the APM_CNT register to cause an SMI#.			RW
04	SLP_SMI_EN	0 = Disables the generation of SMI# on SLP_EN. This bit must be 0 before the software attempts to transition the system into a sleep state by writing a 1 to the SLP_EN bit. 1 = A write of 1 to the SLP_EN bit (bit 13 in PM1_CNT register) will generate an SMI#, and the system will not transition to the sleep state based on that write to the SLP_EN bit.			RW
03	LEGACY_USB_EN	0 = Disable. 1 = Enables legacy USB circuit to cause SMI#.			RW
02	BIOS_EN	0 = Disable. 1 = Enables the generation of SMI# when ACPI software writes a 1 to the GBL_RLS bit (B0:D31:F0:PMBase + 04h:bit 2). If the BIOS_STS bit (B0:D31:F0:PMBase + 34h:bit 2), which gets set when software writes 1 to GBL_RLS bit, is already a 1 at the time that BIOS_EN becomes 1, an SMI# will be generated when BIOS_EN gets set.			RW
01	EOS	End of SMI — (special). This bit controls the arbitration of the SMI signal to the processor. This bit must be set for the PCH to assert SMI# low to the processor after SMI# has been asserted previously. 0 = Once the PCH asserts SMI# low, the EOS bit is automatically cleared. 1 = When this bit is set to 1, SMI# signal will be deasserted for 4 PCI clocks before its assertion. In the SMI handler, the processor should clear all pending SMIs (by servicing them and then clearing their respective status bits), set the EOS bit, and exit SMM. This will allow the SMI arbiter to re-assert SMI upon detection of an SMI event and the setting of a SMI status bit. Note: PCH is able to generate first SMI after reset even though EOS bit is not set. Subsequent SMI require EOS bit is set.			RW
00	GBL_SMI_EN	0 = No SMI# will be generated by PCH. This bit is reset by a PCI reset event. 1 = Enables the generation of SMI# in the system upon any enabled SMI event. When the SMI_LOCK bit is set, this bit cannot be changed.			RW

6.7.3.8 Offset PMBASE + 34h: SMI_STS—SMI Status Register

Note: If the corresponding _EN bit is set when the _STS bit is set, the PCH will cause an SMI# (except bits 8–10 and 12, which do not need enable bits since they are logic ORs of other registers that have enable bits). The PCH uses the same GPE0_EN register (I/O address: PMBase+2Ch) to enable/disable both SMI and ACPI SCI general purpose input events. ACPI OS assumes that it owns the entire GPE0_EN register per ACPI spec. Problems arise when some of the general-purpose inputs are enabled as SMI by BIOS, and some of the general purpose inputs are enabled for SCI. In this case ACPI OS turns off the enabled bit for any GPIx input signals that are not indicated as SCI general-purpose events at boot, and exit from sleeping states. BIOS should define a dummy control method which prevents the ACPI OS from clearing the SMI GPE0_EN bits.

Table 6-104. Offset PMBASE + 34h: SMI_STS—SMI Status Register (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 34h **Offset End:** 34h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
27	GPIO_UNLOCK_SMI_STS	This bit will be set if the GPIO registers lockdown logic is requesting an SMI#. Writing a '1' to this bit position clears this bit to '0'.			RWC
26	SPI_STS	This bit will be set if the SPI logic is generating an SMI#. This bit is read only because the sticky status and enable bits associated with this function are located in the SPI registers.			RO
25 :22	Reserved	Reserved			
21	MONITOR_STS	This bit will be set if the Trap/SMI logic has caused the SMI. This will occur when the processor or a bus master accesses an assigned register (or a sequence of accesses).			RO
20	PCI_EXP_SMI_STS	PCI Express* SMI event occurred. This could be due to a PCI Express* PME event or Hot-Plug event.			RO
19	Reserved	Reserved			
18	INTEL_USB2_STS	This non-sticky read-only bit is a logical OR of each of the SMI status bits in the Intel-Specific USB2 SMI Status Register ANDed with the corresponding enable bits. Additionally, the Port Disable Write Enable SMI is reported in this bit; the specific status bit for this event is contained in the USB Per-Port Registers Write Control Register in this I/O space. This bit will not be active if the enable bits are not set. Writes to this bit will have no effect. All integrated USB2 Host Controllers are represented with this bit.			RO
17	LEGACY_USB2_STS	This non-sticky read-only bit is a logical OR of each of the SMI status bits in the USB2 Legacy Support Register ANDed with the corresponding enable bits. This bit will not be active if the enable bits are not set. Writes to this bit will have no effect. All integrated USB2 Host Controllers are represented with this bit.			RO

Table 6-104. Offset PMBASE + 34h: SMI_STS—SMI Status Register (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 34h **Offset End:** 34h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
16	SMBUS_SMI_STS	SMBus SMI Status — Software clears this bit by writing a 1 to it. 0 = This bit is set from the 64 kHz clock domain used by the SMBus. Software must wait at least 15.63 µs after the initial assertion of this bit before clearing it. 1 = Indicates that the SMI# was caused by: - The SMBus Slave receiving a message that an SMI# should be caused, or - The SMBALERT# signal goes active and the SMB_SMI_EN bit is set and the SMBALERT_DIS bit is cleared, or - The SMBus Slave receiving a Host Notify message and the HOST_NOTIFY_INTREN and the SMB_SMI_EN bits are set, or - The PCH detecting the SMLINK_SLAVE_SMI command while in the S0 state.			RWC
15	SERIRQ_SMI_STS	0 = SMI# was not caused by the SERIRQ decoder. 1 = Indicates that the SMI# was caused by the SERIRQ decoder. This is not a sticky bit			RO
14	PERIODIC_STS	Software clears this bit by writing a 1 to it. 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set at the rate determined by the PER_SMI_SEL bits. If the PERIODIC_EN bit (PMBASE + 30h, bit 14) is also set, the PCH generates an SMI#.			RWC
13	TCO_STS	Software clears this bit by writing a 1 to it. 0 = SMI# not caused by TCO logic. 1 = Indicates the SMI# was caused by the TCO logic. This is not a wake event.			RWC
12	DEVMON_STS	Device Monitor Status: 0 = SMI# not caused by Device Monitor. 1 = Set if bit 0 of the DEVACT_STS register (PMBASE + 44h) is set. The bit is not sticky, so writes to this bit will have no effect.			RO
11	MCSMI_STS	Microcontroller SMI# Status — R/WC. Software clears this bit by writing a 1 to it. 0 = indicates that there has been no access to the power management microcontroller range (62h or 66h). 1 = Set if there has been an access to the power management microcontroller range (62h or 66h) and the Microcontroller Decode Enable #1 bit in the LPC Bridge I/O Enables configuration register is 1 (B0:D31:F0:Offset 82h:bit 11). This implementation assumes that the Microcontroller is on LPC. If this bit is set, and the MCSMI_EN bit is also set, the PCH will generate an SMI#.			RWC
10	GPE0_STS	This bit is a logical OR of the bits in the ALT_GP_SMI_STS register that are also set up to cause an SMI# (as indicated by the GPI_ROUT registers) and have the corresponding bit set in the ALT_GP_SMI_EN register. Bits that are not routed to cause an SMI# will have no effect on this bit. 0 = SMI# was not generated by a GPI assertion. 1 = SMI# was generated by a GPI assertion.			RO

Table 6-104. Offset PMBASE + 34h: SMI_STS—SMI Status Register (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 34h **Offset End:** 34h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	GPE0_STS	This bit is a logical OR of the bits 47:32, 14:10, 8, 6:2 and 0 in the GPE0_STS register (PMBASE + 28h) that also have the corresponding bit set in the GPE0_EN register (PMBASE + 2Ch). 0 = SMI# was not generated by a GPE0 event. 1 = SMI# was generated by a GPE0 event.			RO
08	PM1_STS_REG	This is an ORs of the bits in the ACPI PM1 Status Register (offset PMBASE+00h) that can cause an SMI#. 0 = SMI# was not generated by a PM1_STS event. 1 = SMI# was generated by a PM1_STS event.			RO
07	Reserved	Reserved			
06	SWSMI_TMR_STS	Software clears this bit by writing a 1 to it. 0 = Software SMI# Timer has Not expired. 1 = Set by the hardware when the Software SMI# Timer expires.			RWC
05	APM_STS	Software clears this bit by writing a 1 to it. 0 = No SMI# generated by write access to APM Control register with APMCH_EN bit set. 1 = SMI# was generated by a write access to the APM Control register with the APMC_EN bit set.			RWC
04	SLP_SMI_STS	Software clears this bit by writing a 1 to the bit location. 0 = No SMI# caused by write of 1 to SLP_EN bit when SLP_SMI_EN bit is also set. 1 = Indicates an SMI# was caused by a write of 1 to SLP_EN bit when SLP_SMI_EN bit is also set.			RWC
03	LEGACY_USB_STS	This bit is a logical OR of each of the SMI status bits in the USB* Legacy Keyboard/Mouse Control Registers ANDed with the corresponding enable bits. This bit will not be active if the enable bits are not set. 0 = SMI# was not generated by USB* Legacy event. 1 = SMI# was generated by USB* Legacy event.			RO
02	BIOS_STS	0 = No SMI# generated due to ACPI software requesting attention. 1 = This bit gets set by hardware when a 1 is written by software to the GBL_RLS bit (B0:D31:F0:PMBase + 04h:bit 2). When both the BIOS_EN bit (B0:D31:F0:PMBase + 30h:bit 2) and the BIOS_STS bit are set, an SMI# will be generated. The BIOS_STS bit is cleared when software writes a 1 to its bit position.			RWC
01 :00	Reserved	Reserved			

6.7.3.9 Offset PMBASE +38h: ALT_GP_SMI_EN—Alternate GPI SMI Enable Register

Table 6-105. Offset PMBASE +38h: ALT_GP_SMI_EN—Alternate GPI SMI Enable Register

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 38h **Offset End:** 38h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	ALT_GP_SMI_EN	Alternate GPI SMI Enable — These bits are used to enable the corresponding GPIO to cause an SMI#. For these bits to have any effect, the following must be true. - The corresponding bit in the ALT_GP_SMI_EN register is set. - The corresponding GPI must be routed in the GPI_ROUT register to cause an SMI. - The corresponding GPIO must be implemented. Mapping is as follows: bit 15 corresponds to GPIO15... bit 0 corresponds to GPIO0.			RW

6.7.3.10 Offset PMBASE +3Ah: ALT_GP_SMI_STS—Alternate GPI SMI Status Register

Table 6-106. Offset PMBASE +3Ah: ALT_GP_SMI_STS—Alternate GPI SMI Status Register

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 3Ah **Offset End:** 3Ah	
Size: 16 bit	**Default:** 0000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	ALT_GP_SMI_STS	Alternate GPI SMI Status — These bits report the status of the corresponding GPIOs. 0 = Inactive. Software clears this bit by writing a 1 to it. 1 = Active These bits are sticky. If the following conditions are true, then an SMI# will be generated and the GPE0_STS bit set: - The corresponding bit in the ALT_GPI_SMI_EN register (PMBASE + 38h) is set - The corresponding GPIO must be routed in the GPI_ROUT register to cause an SMI. - The corresponding GPIO must be implemented. All bits are in the resume well. Default for these bits is dependent on the state of the GPIO pins.			RWC

6.7.3.11 Offset PMBASE +3Ch: UPRWC—USB Per-Port Registers Write Control

Table 6-107. Offset PMBASE +3Ch: UPRWC—USB Per-Port Registers Write Control

Description:					
View: PCI	**BAR:** PMBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 16 bit	**Default:** 0000h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved			
08		Write Enable Status: 0 = This bit gets set by hardware when the "Per-Port Registers Write Enable" bit is written from 0 to 1 1 = This bit is cleared by software writing a 1b to this bit location The setting condition takes precedence over the clearing condition in the event that both occur at once. When this bit is 1b and bit 0 is 1b, the INTEL_USB2_STS bit is set in the SMI_STS register.			RWC
07 :02	Reserved	Reserved			
01		USB* Per-Port Registers Write Enable: 0 = Disable 1 = Writes are enabled to the USB* Port Disable Override register (Chipset Config - Offset 3518h)			RW
00		Write Enable SMI Enable: 0 = Disable 1 = enables the generation of SMI when the Per-Port Registers Write Enable (bit 1) is written from 0 to 1. Once written to 1b, this bit can not be cleared by software.			RWO

6.7.3.12 Offset PMBASE +42h: GPE_CNTL—General Purpose Control Register

Table 6-108. Offset PMBASE +42h: GPE_CNTL—General Purpose Control Register

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 42h **Offset End:** 42h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved			
02	GPIO27_POL	This bit controls the polarity of the GPIO27 pin needed to set the GPIO27_STS bit. 0 = GPIO27 - 0 will set the GPIO27_STS bit. 1 = GPIO27 - 1 will set the GPIO27_STS bit.			RW
01	SWGPE_CTRL	This bit allows software to control the assertion of SWGPE_STS bit. This bit is used by hardware as the level input signal for the SWGPE_STS bit in the GPE0_STS register. When SWGPE_CTRL is 1, SWGPE_STS will be set to 1, and writes to SWGPE_STS with a value of 1 to clear SWGPE_STS results in SWGPE_STS being set back to 1 by hardware. When SWGPE_CTRL is 0, writes to SWGPE_STS with a value of 1 will clear SWGPE_STS to 0. This bit is cleared to 0 based on a Power Button Override, CPU Thermal Event as well as by the RSMRST# pin assertion.			RW
00	Reserved	Reserved			

6.7.3.13 Offset PMBASE +44h: DEVACT_STS—Device Activity Status Register

Each bit indicates if an access has occurred to the corresponding device's trap range, or for bits 6:9 if the corresponding PCI interrupt is active. This register is used in conjunction with the Periodic SMI# timer to detect any system activity for legacy power management. The periodic SMI# timer indicates if it is the right time to read the DEVACT_STS register (PMBASE + 44h).

Note: Software clears bits that are set in this register by writing a 1 to the bit position.

Table 6-109. Offset PMBASE +44h: DEVACT_STS—Device Activity Status Register

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 44h **Offset End:** 44h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved			
12	KBC_ACT_STS	KBC (60/64h). 0 = Indicates that there has been no access to this device's I/O range. 1 = This device's I/O range has been accessed. Clear this bit by writing a 1 to the bit location.			RWC
11 :10	Reserved	Reserved			
09	PIRQDH_ACT_STS	PIRQ[D or H]. The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.			RWC
08	PIRQCG_ACT_STS	PIRQ[C or G]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.			RWC
07	PIRQBF_ACT_STS	PIRQ[B or F]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.			RWC
06	PIRQAE_ACT_STS	PIRQ[A or E]. 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts has been active. Clear this bit by writing a 1 to the bit location.			RWC
05 :00	Reserved	Reserved			

6.7.3.14 Offset PMBASE +50h: PM2_CNT—Power Management 2 Control

Table 6-110. Offset PMBASE +50h: PM2_CNT—Power Management 2 Control

Description:					
View: PCI	**BAR:** PMBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 50h **Offset End:** 50h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved			
00	ARB_DIS	Arbiter Disable — This bit is a scratchpad bit for legacy software compatibility.			RW

6.8 System Management TCO Registers (B0:D31:F0)

The TCO logic is accessed via registers mapped to the PCI configuration space (Device 31:Function 0) and the system I/O space. For TCO PCI Configuration registers, See LPC Device 31:Function 0 PCI Configuration registers.

6.8.1 TCO Register I/O Map

The TCO I/O registers reside in a 32-byte range pointed to by a TCOBASE value, which is, PMBASE + 60h in the PCI config space. The following table shows the mapping of the registers within that 32-byte range. Each register is described in the following sections.

Table 6-111. TCO I/O Register Address Map

Offset Start	**Offset End**	**Register ID - Description**	**Default Value**
00h	00h	"Offset TCOBASE +00h: TCO_RLD—TCO Timer Reload and Current Value Register" on page 343	0000h
02h	02h	"Offset TCOBASE +02h: TCO_DAT_IN—TCO Data In Register" on page 343	00h
03h	03h	"Offset TCOBASE +03h: TCO_DAT_OUT—TCO Data Out Register" on page 344	00h
04h	04h	"Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register" on page 344	2000h
06h	06h	"Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register" on page 346	0000h
08h	08h	"Offset TCOBASE +08h: TCO1_CNT—TCO1 Control Register" on page 348	0000h
0Ah	0Ah	"Offset TCOBASE +0Ah: TCO2_CNT—TCO2 Control Register" on page 349	0008h
0Ch	0Dh	"Offset TCOBASE +0Ch and Offset TCOBASE +0Dh: TCO_MESSAGE1 and TCO_MESSAGE2 Registers" on page 350	00h
0Eh	0Eh	"Offset TCOBASE + 0Eh: TCO_WDCNT—TCO Watchdog Control Register" on page 350	00h
10h	10h	"Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register" on page 351	03h
12h	12h	"Offset TCOBASE + 12h:TCO_TMR—TCO Timer Initial Value Register" on page 351	0004h

6.8.1.1 Offset TCOBASE +00h: TCO_RLD—TCO Timer Reload and Current Value Register

Table 6-112. Offset TCOBASE +00h: TCO_RLD—TCO Timer Reload and Current Value Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 00h **Offset End:** 00h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09 :00	TCO_RLD	TCO Timer Value — Reading this register will return the current count of the TCO timer. Writing any value to this register will reload the timer to prevent the timeout.			RW

6.8.1.2 Offset TCOBASE +02h: TCO_DAT_IN—TCO Data In Register

Table 6-113. Offset TCOBASE +02h: TCO_DAT_IN—TCO Data In Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TCO_DAT_IN	TCO Data In Value — This data register field is used for passing commands from the OS to the SMI handler. Writes to this register will cause an SMI and set the SW_TCO_SMI bit in the TCO1_STS register (B0:D31:F0:04h).			RW

6.8.1.3 Offset TCOBASE +03h: TCO_DAT_OUT—TCO Data Out Register

Table 6-114. Offset TCOBASE +03h: TCO_DAT_OUT—TCO Data Out Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)	**Bus:Device:Function :** B0:D31 :F0		**Offset Start:** 03h **Offset End:** 03h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TCO_DAT_OUT	TCO Data Out Value — This data register field is used for passing commands from the SMI handler to the OS. Writes to this register will set the TCO_INT_STS bit in the TCO_STS register. It will also cause an interrupt, as selected by the TCO_INT_SEL bits.			RW

6.8.1.4 Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register

Table 6-115. Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TCOBASE (IO)	**Bus:Device:Function :** B0:D31 :F0		**Offset Start:** 04h **Offset End:** 04h	
Size: 16 bit	**Default:** 2000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13	TCO_SLVSEL	TCO Slave Select — This register bit is Read Only by Host and indicates the value of TCO Slave Select Soft Strap. See the PCH Soft Straps section of the SPI Chapter for details.			RO
12	DMISERR_STS	0 = Software clears this bit by writing a 1 to it. 1 = PCH received a DMI special cycle message via DMI indicating that it wants to cause an SERR#. The software must read the Processor to determine the reason for the SERR#.			RWC
11	Reserved	Reserved			
10	DMISMI_STS	0 = Software clears this bit by writing a 1 to it. 1 = PCH received a DMI special cycle message via DMI indicating that it wants to cause an SMI. The software must read the Processor to determine the reason for the SMI.			RWC
09	DMISCI_STS	0 = Software clears this bit by writing a 1 to it. 1 = PCH received a DMI special cycle message via DMI indicating that it wants to cause an SCI. The software must read the Processor to determine the reason for the SCI.			RWC

Table 6-115. Offset TCOBASE +04h: TCO1_STS—TCO1 Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function :** B0:D31 :F0	**Offset Start:** 04h **Offset End:** 04h	
Size: 16 bit	**Default:** 2000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
08	BIOSWR_STS	0 = Software clears this bit by writing a 1 to it. 1 = PCH sets this bit and generates and SMI# to indicate an invalid attempt to write to the BIOS. This occurs when either: a) The BIOSWP bit is changed from 0 to 1 and the BLD bit is also set, or b) any write is attempted to the BIOS and the BIOSWP bit is also set. Note: On write cycles attempted to the 4 MB lower alias to the BIOS space, the BIOSWR_STS will not be set.			RWC
07	NEWCENTURY_STS	This bit is in the RTC well. 0 = Cleared by writing a 1 to the bit position or by RTCRST# going active. 1 = This bit is set when the Year byte (RTC I/O space, index offset 09h) rolls over from 99 to 00. Setting this bit will cause an SMI# (but not a wake event). Note: The NEWCENTURY_STS bit is not valid when the RTC battery is first installed (or when RTC power has not been maintained). Software can determine if RTC power has not been maintained by checking the RTC_PWR_STS bit (B0:D31:F0:A4h, bit 2), or by other means (such as a checksum on RTC RAM). If RTC power is determined to have not been maintained, BIOS should set the time to a valid value and then clear the NEWCENTURY_STS bit. The NEWCENTURY_STS bit may take up to 3 RTC clocks for the bit to be cleared after a 1 is written to the bit to clear it. After writing a 1 to this bit, software should not exit the SMI handler until verifying that the bit has actually been cleared. This will ensure that the SMI is not re-entered.			RWC
06 :04	Reserved	Reserved			
03	TIMEOUT	0 = Software clears this bit by writing a 1 to it. 1 = Set by PCH to indicate that the SMI was caused by the TCO timer reaching 0.			RWC
02	TCO_INT_STS	0 = Software clears this bit by writing a 1 to it. 1 = SMI handler caused the interrupt by writing to the TCO_DAT_OUT register (TCOBASE + 03h).			RWC
01	SW_TCO_SMI	0 = Software clears this bit by writing a 1 to it. 1 = Software caused an SMI# by writing to the TCO_DAT_IN register (TCOBASE + 02h).			RWC
00	NMI2SMI_STS	0 = Cleared by clearing the associated NMI status bit. 1 = Set by the PCH when an SMI# occurs because an event occurred that would otherwise have caused an NMI (because NMI2SMI_EN is set).			RO

6.8.1.5 Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register

Table 6-116. Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TCOBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 06h **Offset End:** 06h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Resume (Except Bit 0, in RTC)	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :05	Reserved	Reserved			
04	SMLINK_SLV_SMI_STS	SMLink Slave SMI Status — Allow the software to go directly into pre-determined sleep state. This avoids race conditions. Software clears this bit by writing a 1 to it: The bit is reset by RSMRST#, but not due to the PCI Reset associated with exit from S3–S5 states. 1 = PCH sets this bit to 1 when it receives the SMI message on the SMLink's Slave Interface.			RWC
03	Reserved	Reserved			
02	BOOT_STS	0 = Cleared by PCH based on RSMRST# or by software writing a 1 to this bit. Software should first clear the SECOND_TO_STS bit before writing a 1 to clear the BOOT_STS bit. 1 = Set to 1 when the SECOND_TO_STS bit goes from 0 to 1 and the processor has not fetched the first instruction. If rebooting due to a second TCO timer timeout, and if the BOOT_STS bit is set, the PCH will reboot using the 'safe' multiplier (1111). This allows the system to recover from a processor frequency multiplier that is too high, and allows the BIOS to check the BOOT_STS bit at boot. If the bit is set and the frequency multiplier is 1111, then the BIOS knows that the processor has been programmed to an invalid multiplier.			RWC
01	SECOND_TO_STS	0 = Software clears this bit by writing a 1 to it, or by a RSMRST#. 1 = PCH sets this bit to 1 to indicate that the TIMEOUT bit had been (or is currently) set and a second timeout occurred before the TCO_RLD register was written. If this bit is set and the NO_REBOOT config bit is 0, then the PCH will reboot the system after the second timeout. The reboot is done by asserting PLTRST#.			RWC

Table 6-116. Offset TCOBASE +06h: TCO2_STS—TCO2 Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TCOBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 06h **Offset End:** 06h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Resume (Except Bit 0, in RTC)	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
00	INTRD_DET	Intruder Detect: 0 = Software clears this bit by writing a 1 to it, or by RTCRST# assertion. 1 = Set by PCH to indicate that an intrusion was detected. This bit is set even if the system is in G3 state. Notes: 1. This bit has a recovery time. After writing a 1 to this bit position (to clear it), the bit may be read back as a 1 for up 65 microseconds before it is read as a 0. Software must be aware of this recovery time when reading this bit after clearing it. 2. If the INTRUDER# signal is active when the software attempts to clear the INTRD_DET bit, the bit will remain as a 1, and the SMI# will be generated again immediately. The SMI handler can clear the INTRD_SEL bits (TCOBASE + 0Ah, bits 2:1), to avoid further SMIs. However, if the INTRUDER# signals goes inactive and then active again, there will not be further SMI's (because the INTRD_SEL bits would select that no SMI# be generated). 3. If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET signal will go to a 0 when INTRUDER# input signal goes inactive. This is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.			RWC

6.8.1.6 Offset TCOBASE +08h: TCO1_CNT—TCO1 Control Register

Table 6-117. Offset TCOBASE +08h: TCO1_CNT—TCO1 Control Register

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR: TCOBASE (IO)</td><td colspan="2">Bus:Device:Function: B0:D31:F0</td><td>Offset Start: 08h
Offset End: 08h</td></tr>
<tr><td>Size: 16 bit</td><td colspan="3">Default: 0000h</td><td colspan="2">Power Well: Core</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>15 :13</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>12</td><td>TCO_LOCK</td><td>When set to 1, this bit prevents writes from changing the TCO_EN bit (in offset 30h of Power Management I/O space). Once this bit is set to 1, it can not be cleared by software writing a 0 to this bit location. A core-well reset is required to change this bit from 1 to 0. This bit defaults to 0.</td><td></td><td></td><td>RWLO</td></tr>
<tr><td>11</td><td>TCO_TMR_HLT</td><td>TCO Timer Halt:
0 = The TCO Timer is enabled to count.
1 = The TCO Timer will halt. It will not count, and thus cannot reach a value that will cause an SMI# or set the SECOND_TO_STS bit. When set, this bit will prevent rebooting and prevent Alert On LAN event messages from being transmitted on the SMLink (but not Alert On LAN* heartbeat messages).</td><td></td><td></td><td>RW</td></tr>
<tr><td>10</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>09</td><td>NMI2SMI_EN</td><td>0 = Normal NMI functionality.
1 = Forces all NMIs to instead cause SMIs. The functionality of this bit is dependent upon the settings of the NMI_EN bit and the GBL_SMI_EN bit as detailed in the following table:
<table><tr><th>NMI_EN</th><th>GBL_SMI_EN</th><th>Description</th></tr><tr><td>0b</td><td>0b</td><td>No SMI# at all because GBL_SMI_EN = 0</td></tr><tr><td>0b</td><td>1b</td><td>SMI# will be caused due to NMI events</td></tr><tr><td>1b</td><td>0b</td><td>No SMI# at all because GBL_SMI_EN = 0</td></tr><tr><td>1b</td><td>1b</td><td>No SMI# due to NMI because NMI_EN = 1</td></tr></table></td><td></td><td></td><td>RW</td></tr>
<tr><td>08</td><td>NMI_NOW</td><td>0 = Software clears this bit by writing a 1 to it. The NMI handler is expected to clear this bit. Another NMI will not be generated until the bit is cleared.
1 = Writing a 1 to this bit causes an NMI. This allows the BIOS or SMI handler to force an entry to the NMI handler.</td><td></td><td></td><td>RWC</td></tr>
<tr><td>07 :00</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
</table>

6.8.1.7 Offset TCOBASE +0Ah: TCO2_CNT—TCO2 Control Register

Table 6-118. Offset TCOBASE +0Ah: TCO2_CNT—TCO2 Control Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 16 bit	**Default:** 0008h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :06	Reserved	Reserved			
05 :04	OS_POLICY	OS-based software writes to these bits to select the policy that the BIOS will use after the platform resets due the WDT. The following convention is recommended for the BIOS and OS: 00 = Boot normally 01 = Shut down 10 = Don't load OS. Hold in pre-boot state and use LAN to determine next step 11 = Reserved These are scratchpad bits. They should not be reset when the TCO logic resets the platform due to Watchdog Timer.			RW
03	GPIO11_ALERT_DISABLE	At reset (via RSMRST# asserted) this bit is set and GPIO[11] alerts are disabled. 0 = Enable. 1 = Disable GPIO11/MST_SMBALERT# as an alert source for the heartbeats and the SMBus slave.			RW
02 :01	INTRD_SEL	This field selects the action to take if the INTRUDER# signal goes active. 00 = No interrupt or SMI# 01 = Interrupt (as selected by TCO_INT_SEL). 10 = SMI 11 = Reserved			RW
00	Reserved	Reserved			

6.8.1.8 Offset TCOBASE +0Ch and Offset TCOBASE +0Dh: TCO_MESSAGE1 and TCO_MESSAGE2 Registers

I/O Address:
TCOBASE +0Ch (Message 1)
TCOBASE +0Dh (Message 2)

Table 6-119. Offset TCOBASE +0Ch and Offset TCOBASE +0Dh: TCO_MESSAGE1 and TCO_MESSAGE2 Registers

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Ch **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TCO_MESSAGE[*n*]	BIOS can write into these registers to indicate its boot progress. The external microcontroller can read these registers to monitor the boot progress			RW

6.8.1.9 Offset TCOBASE + 0Eh: TCO_WDCNT—TCO Watchdog Control Register

Table 6-120. Offset TCOBASE + 0Eh: TCO_WDCNT—TCO Watchdog Control Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TCO_WDCNT	The BIOS or system management software can write into this register to indicate more details on the boot progress. The register will reset to 00h based on a RSMRST# (but not PLTRST#). The external microcontroller can read this register to monitor boot progress.			RW

6.8.1.10 Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register

Table 6-121. Offset TCOBASE + 10h: SW_IRQ_GEN—Software IRQ Generation Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 10h **Offset End:** 10h	
Size: 8 bit	**Default:** 03h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :02	Reserved	Reserved			
01	IRQ12_CAUSE	When software sets this bit to 1, IRQ12 will be asserted. When software sets this bit to 0, IRQ12 will be deasserted.			RW
00	IRQ1_CAUSE	When software sets this bit to 1, IRQ1 will be asserted. When software sets this bit to 0, IRQ1 will be deasserted.			RW

6.8.1.11 Offset TCOBASE + 12h:TCO_TMR—TCO Timer Initial Value Register

Table 6-122. Offset TCOBASE + 12h:TCO_TMR—TCO Timer Initial Value Register

Description:					
View: PCI	**BAR:** TCOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 12h **Offset End:** 12h	
Size: 16 bit	**Default:** 0004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09 :00		TCO Timer Initial Value — Value that is loaded into the timer each time the TCO_RLD register is written. Values of 0000h or 0001h will be ignored and should not be attempted. The timer is clocked at approximately 0.6 seconds, and thus allows timeouts ranging from 1.2 second to 613.8 seconds. The timer has an error of ±1 tick (0.6s). The TCO Timer will only count down in the S0 state.			RW

6.9 General Purpose I/O Registers (B0:D31:F0)

6.9.1 General Purpose I/O Signals

The control for the general purpose I/O signals is handled through a 128-byte I/O space. The base offset for this space is selected by the GPIOBASE register.

Table 6-123. General Purpose I/O Signals

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register" on page 353	F96BA1FFh
04h	07h	"Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register" on page 354	F6FF6EFFh
0Ch	0Fh	"Offset GPIOBASE + 0Ch:GP_LVL—GPIO Level for Input or Output Register" on page 354	02FE0100h
18h	1Bh	"Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register" on page 355	00040000h
1Ch	1Fh	"Offset GPIOBASE + 1Ch: GP_SER_BLINK—GP Serial Blink Data" on page 356	00000000h
20h	23h	"Offset GPIOBASE + 20h: GP_SB_CMDSTS—GP Serial Blink Command Status" on page 357	00080000h
24h	27h	"Offset GPIOBASE + 24h: GP_SB_DATA—GP Serial Blink Data" on page 358	00000000h
28h	29h	"Offset GPIOBASE + 28h: GPI_NMI_EN—GPI NMI Enable" on page 358	00000h
2Ah	2Bh	"Offset GPIOBASE + 2Ah: GPI_NMI_STS—GPI NMI Status" on page 359	00000h
2Ch	2Fh	"Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register" on page 359	00000000h
30h	33h	"Offset GPIOBASE + 30h: GPIO_USE_SEL2—GPIO Use Select 2 Register" on page 360	020300FEh
34h	37h	"Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register" on page 361	1F57FFF4h
38h	3Bh	"Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register" on page 361	A4AA0003h
40h	43h	"Offset GPIOBASE + 40h: GPIO_USE_SEL3—GPIO Use Select 3 Register" on page 362	00000000h
44h	47h	"Offset GPIOBASE + 44h: GP_IO_SEL3—GPIO Input/Output Select 3 Register" on page 363	00000F00
48h	4Bh	"Offset GPIOBASE + 48h: GP_LVL3—GPIO Level for Input or Output 3 Register" on page 364	00000000h
60h	63h	"Offset GPIOBASE + 60h: GP_RST_SEL1—GPIO Reset Select" on page 365	01000000h
64h	67h	"Offset GPIOBASE + 64h: GP_RST_SEL2—GPIO Reset Select" on page 366	00000000h
68h	6Bh	"Offset GPIOBASE + 68h: GP_RST_SEL3—GPIO Reset Select" on page 366	00000000h

6.9.1.1 Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register

Table 6-124. Offset GPIOBASE + 00h: GPIO_USE_SEL—GPIO Use Select Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** F96BA1FFh			**Power Well:**	Core for 0:7, 16:23, Resume for 8:15, 24:3
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPIO_USE_SEL[31:0]	Enables GPIO[n] (where n is the bit number) to be used as a GPIO, rather than for the native function. 1 = Signal used as GPIO (or unmuxed). 0 = Signal used as native function.(or unimplemented GPIO). Unmuxed GPIOs must report 1b in this register. Unimplemented GPIOs must report 0b in this register. If GPIO vs Native Mode are configured via SPI Soft Strap, the corresponding GPIO_USE_SEL bits for these GPIOs have no effect. Notes: 1. The following bits are always 1 because they are always unMultiplexed: 0, 8, 15, 24, 27, and 28. 2. If GPIO[n] does not exist, then, the n-bit in this register will always read as 0 and writes will have no effect. 3. After a full reset (RSMRST#) all multiplexed signals in the resume and core wells are configured as their default function. After only a PLTRST#, the GPIOs in the core well are configured as their default function. 4. When configured to GPIO mode, the muxing logic will present the inactive state to native logic that uses the pin as an input. 5. All GPIOs are reset to the default state by CF9h reset except GPIO24. 6. Bit 26 may be overridden by bit 8 in the GEN_PMCON_3 Register.			RW

6.9.1.2 Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register

Table 6-125. Offset GPIOBASE + 04h: GP_IO_SEL—GPIO Input/Output Select Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 04h **Offset End:** 07h	
Size: 16 bit	**Default:** F6FF6EFFh			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_IO_SEL [31:0]	When configured in native mode (GPIO_USE_SEL[n] is 0), writes to these bits have no effect. The value reported in this register is undefined when programmed as native mode. 0 = Output. The corresponding GPIO signal is an output. 1 = Input. The corresponding GPIO signal is an input.			RW

6.9.1.3 Offset GPIOBASE + 0Ch:GP_LVL—GPIO Level for Input or Output Register

Table 6-126. Offset GPIOBASE + 0Ch:GP_LVL—GPIO Level for Input or Output Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 02FE0100h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_LVL [31:0]	These registers are implemented as dual read/write with dedicated storage each. Write value will be stored in the write register, while read is coming from the read register which will always reflect the value of the pin. If GPIO[n] is programmed to be an output (via the corresponding bit in the GP_IO_SEL register), then the corresponding GP_LVL[n] write register value will drive a high or low value on the output pin. 1 = high, 0 = low. When configured in native mode (GPIO_USE_SEL[n] is 0), writes to these bits are stored but have no effect to the pin value. The value reported in this register is undefined when programmed as native mode.			RW

6.9.1.4 Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register

Table 6-127. Offset GPIOBASE + 18h: GPO_BLINK—GPO Blink Enable Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00040000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 00	GP_BLINK [31:0]	The setting of this bit has no effect if the corresponding GPIO signal is programmed as an input. 0 = The corresponding GPIO will function normally. 1 = If the corresponding GPIO is programmed as an output, the output signal will blink at a rate of approximately once per second. The high and low times have approximately 0.5 seconds each. The GP_LVL bit is not altered when this bit is set. The value of the corresponding GP_LVL bit remains unchanged during the blink process, and does not effect the blink in any way. The GP_LVL bit is not altered when programmed to blink. It will remain at its previous value. These bits correspond to GPIO in the Resume well. These bits revert to the default value based on RSMRST# or a write to the CF9h register (but not just on PLTRST#).			RW

Note: GPIO18 will toggle at a frequency of approximately 1Hz when the signal is programmed as a GPIO (configured as an output via the GP_IO_SEL register).

6.9.1.5 Offset GPIOBASE + 1Ch: GP_SER_BLINK—GP Serial Blink Data

Table 6-128. Offset GPIOBASE + 1Ch: GP_SER_BLINK—GP Serial Blink Data

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_SER_BLINK [31:0]	The setting of this bit has no effect if the corresponding GPIO is programmed as an input or if the corresponding GPIO has the GPO_BLINK bit set. When set to a '0', the corresponding GPIO will function normally. When using serial blink, this bit should be set to a 1 while the corresponding GP_IO_SEL bit is set to 1. Setting the GP_IO_SEL bit to 0 after the GP_SER_BLINK bit ensures PCH will not drive a 1 on the pin as an output. When this corresponding bit is set to a 1 and the pin is configured to output mode, the serial blink capability is enabled. The PCH will serialize messages through an open-drain buffer configuration. The value of the corresponding GP_LVL bit remains unchanged and does not impact the serial blink capability in any way. Writes to this register have no effect when the corresponding pin is configured in native mode and the read value returned is undefined.			RW

6.9.1.6 Offset GPIOBASE + 20h: GP_SB_CMDSTS—GP Serial Blink Command Status

Table 6-129. Offset GPIOBASE + 20h: GP_SB_CMDSTS—GP Serial Blink Command Status

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00080000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :22	DLS	Data Length Select — This field determines the number of bytes to serialize on GPIO 00 = Serialize bits 7:0 of GP_SB_DATA (1 byte) 01 = Serialize bits 15:0 of GP_SB_DATA (2 bytes) 10 = Undefined - Software must not write this value 11 = Serialize bits 31:0 of GP_SB_DATA (4 bytes) Software should not modify the value in this register unless the Busy bit is clear. Writes to this register have no effect when the corresponding pin is configured in native mode and the read value returned is undefined.			RW
21 :16	DRS	Data Rate Select — This field selects the number of 120ns time intervals to count between Manchester data transitions. The default of 8h results in a 960ns minimum time between transitions. A value of 0h in this register produces undefined behavior. Software should not modify the value in this register unless the Busy bit is clear.			RW
15 :09	Reserved	Reserved			
08		Busy — This read-only status bit is the hardware indication that a serialization is in progress. Hardware sets this bit to 1 based on the Go bit being set. Hardware clears this bit when the Go bit is cleared by the hardware.			RO
07 :01	Reserved	Reserved			
00		Go — This bit is set to 1 by software to start the serialization process. Hardware clears the bit after the serialized data is sent. Writes of 0 to this register have no effect. Software should not write this bit to 1 unless the Busy status bit is cleared.			RW

6.9.1.7 Offset GPIOBASE + 24h: GP_SB_DATA—GP Serial Blink Data

Table 6-130. Offset GPIOBASE + 24h: GP_SB_DATA—GP Serial Blink Data

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_SB_DATA [31:0]	This register contains the data serialized out. The number of bits shifted out are selected through the DLS field in the GP_SB_CMDSTS register. This register should not be modified by software when the Busy bit is set.			RW

6.9.1.8 Offset GPIOBASE + 28h: GPI_NMI_EN—GPI NMI Enable

Table 6-131. Offset GPIOBASE + 28h: GPI_NMI_EN—GPI NMI Enable

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 28h **Offset End:** 29h	
Size: 16 bit	**Default:** 00000h			**Power Well:** Core for 07:00 Resume for 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	GPI_NMI_EN [15:0]	GPI NMI Enable: This bit only has effect if the corresponding GPIO is used as an input and its GPI_ROUT register is being programmed to NMI functionality. When set to '1', it used to allow active-low and active-high inputs (depends on inversion bit) to cause NMI.			RW

6.9.1.9 Offset GPIOBASE + 2Ah: GPI_NMI_STS—GPI NMI Status

Table 6-132. Offset GPIOBASE + 2Ah: GPI_NMI_STS—GPI NMI Status

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2Ah **Offset End:** 2Bh	
Size: 16 bit	**Default:** 00000h			**Power Well:** Core for 07:00 Resume for 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	GPI_NMI_STS [15:0]	GPI NMI Status: This bit is set if the corresponding GPIO is used as an input, and its GPI_ROUT register is being programmed to NMI functionality and also GPI_NMI_EN bit is set when it detects either: 1) active-high edge when its corresponding GPI_INV is configured with value 0. 2) active-low edge when its corresponding GPI_INV is configured with value 1. Writing value of 1 will clear the bit, while writing value of 0 have no effect.			RW

6.9.1.10 Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register

Table 6-133. Offset GPIOBASE + 2Ch: GPI_INV—GPIO Signal Invert Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2Ch **Offset End:** 2Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** CPU I/O for 17, Core for 16, 07:00	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_INV[n]	Input Inversion — This bit only has effect if the corresponding GPIO is used as an input and used by the GPE logic, where the polarity matters. When set to '1', then the GPI is inverted as it is sent to the GPE logic that is using it. This bit has no effect on the value that is reported in the GP_LVL register. These bits are used to allow both active-low and active-high inputs to cause SMI# or SCI. In the S0 or S1 state, the input signal must be active for at least two PCI clocks to ensure detection by the PCH. In the S3, S4 or S5 states the input signal must be active for at least 2 RTC clocks to ensure detection. The setting of these bits has no effect if the corresponding GPIO is programmed as an output. These bits correspond to GPI that are in the resume well, and will be reset to their default values by RSMRST# or by a write to the CF9h register. 0 = The corresponding GPI_STS bit is set when the PCH detects the state of the input pin to be high. 1 = The corresponding GPI_STS bit is set when the PCH detects the state of the input pin to be low.			RW

6.9.1.11 Offset GPIOBASE + 30h: GPIO_USE_SEL2—GPIO Use Select 2 Register

Table 6-134. Offset GPIOBASE + 30h: GPIO_USE_SEL2—GPIO Use Select 2 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 30h **Offset End:** 33h	
Size: 32 bit	**Default:** 020300FEh			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPIO_USE_SEL2 [63:32]	Each bit in this register enables the corresponding GPIO (if it exists) to be used as a GPIO, rather than for the native function. 0 = Signal used as native function. 1 = Signal used as a GPIO. ***Notes:*** 1. Bits[3, 25] are always 1 because they are unMultiplexed 2. If GPIO[n] does not exist, then, the (n-32) bit in this register will always read as 0 and writes will have no effect 3. After a full reset RSMRST# all multiplexed signals in the resume and core wells are configured as their default function. After only a PLTRST#, the GPIOs in the core well are configured as their default function. 4. When configured to GPIO mode, the muxing logic will present the inactive state to native logic that uses the pin as an input. 5. Bit 26 is ignored, functionality is configured by bits 9:8 of FLMAP0 register. 6. This register corresponds to GPIO[63:32]. Bit 0 corresponds to GPIO32 and bit 31 corresponds to GPIO63.			RW

6.9.1.12 Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register

Table 6-135. Offset GPIOBASE + 34h: GP_IO_SEL2—GPIO Input/Output Select 2 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 34h **Offset End:** 37h	
Size: 32 bit	**Default:** 1F57FFF4h			**Power Well:**	Core for 23:16, 07:00 Resume for 31:24, 15:08
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_IO_SEL2[63:32]	0 = GPIO signal is programmed as an output. 1 = Corresponding GPIO signal (if enabled in the GPIO_USE_SEL2 register) is programmed as an input. This register corresponds to GPIO[63:32]. Bit 0 corresponds to GPIO32.			RW

6.9.1.13 Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register

Table 6-136. Offset GPIOBASE + 38h: GP_LVL2—GPIO Level for Input or Output 2 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 38h **Offset End:** 3Bh	
Size: 32 bit	**Default:** A4AA0003h			**Power Well:**	Core for 23:16, 07:00 Resume for 31:24, 15:08
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GP_LVL[63:32]	These registers are implemented as dual read/write with dedicated storage each. Write value will be stored in the write register, while read is coming from the read register which will always reflect the value of the pin. If GPIO[n] is programmed to be an output (via the corresponding bit in the GP_IO_SEL register), then the corresponding GP_LVL[n] write register value will drive a high or low value on the output pin. 1 = high, 0 = low. When configured in native mode (GPIO_USE_SEL[n] is 0), writes to these bits are stored but have no effect to the pin value. The value reported in this register is undefined when programmed as native mode. This register corresponds to GPIO[63:32]. Bit 0 corresponds to GPIO32.			RW

6.9.1.14 Offset GPIOBASE + 40h: GPIO_USE_SEL3—GPIO Use Select 3 Register

Table 6-137. Offset GPIOBASE + 40h: GPIO_USE_SEL3—GPIO Use Select 3 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 40h **Offset End:** 43h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12		Always 0. No corresponding GPIO.			
11 :08	GPIO_USE_SEL3 [75:72]	Each bit in this register enables the corresponding GPIO (if it exists) to be used as a GPIO, rather than for the native function. 0 = Signal used as native function. 1 = Signal used as a GPIO. Notes: 1. The following bit is always 1 because it is always unMultiplexed: 8 2. If GPIO[n] does not exist, then, the (n-32) bit in this register will always read as 0 and writes will have no effect. 3. After a full reset RSMRST# all multiplexed signals in the resume and core wells are configured as their default function. After only a PLTRST#, the GPIOs in the core well are configured as their default function. 4. When configured to GPIO mode, the muxing logic will present the inactive state to native logic that uses the pin as an input. This register corresponds to GPIO75:72]. Bit 8 corresponds to GPIO72 and bit 11 corresponds to GPIO75.			RW
07 :04		Always 0. No corresponding GPIO.			
03 :00	Reserved	Reserved			

6.9.1.15 Offset GPIOBASE + 44h: GP_IO_SEL3—GPIO Input/Output Select 3 Register

Table 6-138. Offset GPIOBASE + 44h: GP_IO_SEL3—GPIO Input/Output Select 3 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:** 00000F00			**Power Well:**	Core for 23:16, 07:00 Resume for 31:24, 15:08
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12		Always 0. No corresponding GPIO.			
11 :08	GPIO_IO_SEL3 [75:72]	0 = GPIO signal is programmed as an output. 1 = Corresponding GPIO signal (if enabled in the GPIO_USE_SEL3 register) is programmed as an input. This register corresponds to GPIO[75:72]. Bit 8 corresponds to GPIO72 and bit 11 corresponds to GPIO75.			RW
07 :04		Always 0. No corresponding GPIO.			
03 :00	Reserved	Reserved			

6.9.1.16 Offset GPIOBASE + 48h: GP_LVL3—GPIO Level for Input or Output 3 Register

Table 6-139. Offset GPIOBASE + 48h: GP_LVL3—GPIO Level for Input or Output 3 Register

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 48h **Offset End:** 4Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12		Always 0. No corresponding GPIO.			
11 :08	GP_LVL[75:72]	These registers are implemented as dual read/write with dedicated storage each. Write value will be stored in the write register, while read is coming from the read register which will always reflect the value of the pin. If GPIO[n] is programmed to be an output (via the corresponding bit in the GP_IO_SEL register), then the corresponding GP_LVL[n] write register value will drive a high or low value on the output pin. 1 = high, 0 = low. When configured in native mode (GPIO_USE_SEL[n] is 0), writes to these bits are stored but have no effect to the pin value. The value reported in this register is undefined when programmed as native mode. This register corresponds to GPIO[75:72]. Bit 0 corresponds to GPIO72 and bit 11 corresponds to GPIO75.			RW
07 :04		Always 0. No corresponding GPIO.			
03 :00	Reserved	Reserved			

6.9.1.17 Offset GPIOBASE + 60h: GP_RST_SEL1—GPIO Reset Select

Table 6-140. Offset GPIOBASE + 60h: GP_RST_SEL1—GPIO Reset Select

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 60h **Offset End:** 63h	
Size: 32 bit	**Default:** 01000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	GP_RST_SEL [31:24]	0 = Corresponding GPIO registers will be reset by PWROK deassertion, CF9h reset (06h or 0Eh), or SYS_RST# assertion. 1 = Corresponding GPIO registers will be reset by RSMRST# assertion only. GPIO[24] register bits are not cleared by CF9h reset by default.			RW
23 :16	Reserved	Reserved			
15 :08	GP_RST_SEL [15:8]	0 = Corresponding GPIO registers will be reset by PWROK deassertion, CF9h reset (06h or 0Eh), or SYS_RST# assertion. 1 = Corresponding GPIO registers will be reset by RSMRST# assertion only.			RW
07 :00	Reserved	Reserved			

6.9.1.18 Offset GPIOBASE + 64h: GP_RST_SEL2—GPIO Reset Select

Table 6-141. Offset GPIOBASE + 64h: GP_RST_SEL2—GPIO Reset Select

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	GP_RST_SEL [63:56]	0 = Corresponding GPIO registers will be reset by PWROK deassertion, CF9h reset (06h or 0Eh), or SYS_RST# assertion. 1 = Corresponding GPIO registers will be reset by RSMRST# assertion only.			RW
23 :16	Reserved	Reserved			
15 :08	GP_RST_SEL [47:40]	0 = Corresponding GPIO registers will be reset by PWROK deassertion, CF9h reset (06h or 0Eh), or SYS_RST# assertion. 1 = Corresponding GPIO registers will be reset by RSMRST# assertion only.			RW
07 :00	Reserved	Reserved			

6.9.1.19 Offset GPIOBASE + 68h: GP_RST_SEL3—GPIO Reset Select

Table 6-142. Offset GPIOBASE + 68h: GP_RST_SEL3—GPIO Reset Select

Description:					
View: PCI	**BAR:** GPIOBASE (IO)		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 68h **Offset End:** 6Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core for 23:16, 07:00 Resume for 31:24, 15:08	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11 :08	GP_RST_ SEL[75:72]	0 = Corresponding GPIO registers will be reset by PWROK deassertion, CF9h reset (06h or 0Eh), or SYS_RST# assertion. 1 = Corresponding GPIO registers will be reset by RSMRST# assertion only.			RW
07 :00	Reserved	Reserved			

§ §

7.0 Chipset Configuration Registers (B0:D31:F0)

This section describes all registers and base functionality that is related to chipset configuration and not a specific interface (such as LPC or PCI Express*). It contains the root complex register block, which describes the behavior of the upstream internal link.

This block is mapped into memory space, using the Root Complex Base Address (RCBA) register of the PCI-to-LPC bridge. Accesses in this space must be limited to 32-bit (DW) quantities. Burst accesses are not allowed.

All Chipset Configuration Registers are located in the core well unless otherwise indicated.

7.1 Chipset Configuration Registers

7.1.1 Chipset Configuration Register Memory Map (Memory Space)

Note: Address locations that are not shown should be treated as Reserved.

Table 7-1. Chipset Configuration Registers (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
0014h	0017h	"Offset 0014h: V0CTL—Virtual Channel 0 Resource Control Register" on page 370	80000023h
001Ah	001Bh	"Offset 001Ah: V0STS—Virtual Channel 0 Resource Status Register" on page 371	0000h
001Ch	001Fh	"Offset 001Ch: V1CAP—Virtual Channel 1 Resource Capability Register" on page 371	00008001h
0020h	0023h	"Offset 0020h: V1CTL—Virtual Channel 1 Resource Control Register" on page 372	00000000h
0026h	0027h	"Offset 0026h: V1STS—Virtual Channel 1 Resource Status Register" on page 372	0000h
0050h	0053h	"Offset 0050h: CIR0—Chipset Initialization Register 0" on page 373	00000000h
0088h	008Bh	"Offset 0088h: CIR1—Chipset Initialization Register 1" on page 373	00000000h
00ACh	00AFh	"Offset 00ACh: REC—Root Error Command Register" on page 374	0000h
01A0h	01A3h	"Offset 01A0h: ILCL—Internal Link Capabilities List Register" on page 374	00010006h
01A4h	01A7h	"Offset 01A4h: LCAP—Link Capabilities Register" on page 375	00012841h
01A8h	01A9h	"Offset 01A8h: LCTL—Link Control Register" on page 375	0000h
01AAh	01ABh	"Offset 01AAh: LSTS—Link Status Register" on page 376	0041h
0220h	0223h	"Offset 0220h: BCR—Backbone Configuration Register" on page 376	00000000h
0224h	0227h	"Offset 0224h: RPC—Root Port Configuration Register" on page 377	0000000yh (y = 00xxb)
0234h	0237h	"Offset 0234h: DMIC—DMI Control Register" on page 378	00000000h
0238h	023Ch	"Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports" on page 378	76543210h
0290h	0293h	"Offset 0290h: Reserved" on page 379	00000000h

Table 7-1. Chipset Configuration Registers (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
1D40h	1D47h	"Offset 1D40H: CIR5—Chipset Initialization Register 5" on page 380	0000000000000000h
1E00h	1E03h	"Offset 1E00h: TRSR—Trap Status Register" on page 380	00000000h
1E10h	1E17h	"Offset 1E10h: TRCR—Trapped Cycle Register" on page 381	0000000000000000h
1E18h	1E1Fh	"Offset 1E18h: TWDR—Trapped Write Data Register" on page 381	0000000000000000h
1E80h	1E9Fh	"Offset 1E80h: IOTRn—I/O Trap Register (0-3)" on page 382	0000000000000000h
2010h	2013h	"Offset 2010h: DMC—DMI Miscellaneous Control Register" on page 383	00000002h
2024h	2027h	"Offset 2024h: CIR6—Chipset Initialization Register 6" on page 383	0B4030C0h
2324h	2327h	"Offset 2324h: DMC2—DMI Miscellaneous Control Register 2" on page 384	0FFF0FFFh
2330h	2333h	"Offset 60h: SBI Unified AFE Address Register (SATA–B0:D31:F2)" on page 384	00000000h
2334h	2337h	"Offset 64h: SBI Unified AFE Data Register (SATA–B0:D31:F2)" on page 385	00000000h
2338h	2339h	"Offset 68h: SBI Unified AFE Status Register (SATA–B0:D31:F2)" on page 386	0000h
3000h	3000h	"Offset 3000h: TCTL—TCO Configuration Register" on page 387	00h
3100h	3103h	"Offset 3100h: D31IP—Device 31 Interrupt Pin Register" on page 388	03243200h
3108h	310Bh	"Offset 3108h: D29IP—Device 29 Interrupt Pin Register" on page 389	10004321h
310Ch	310Fh	"Offset 310Ch: D28IP—Device 28 Interrupt Pin Register" on page 390	00214321h
3124h	3127h	"Offset 3124h: D22IP—Device 22 Interrupt Pin Register" on page 391	See register description
3140h	3127h	"Offset 3140h: D31IR—Device 31 Interrupt Route Register" on page 392	3210h
3144h	3145h	"Offset 3144h: D29IR—Device 29 Interrupt Route Register" on page 393	3210h
3146h	3147h	"Offset 3146h: D28IR—Device 28 Interrupt Route Register" on page 394	3210h
315Ch	315Dh	"Offset 315Ch: D22IR—Device 22 Interrupt Route Register" on page 395	3210h
31FEh	31FFh	"Offset 31FEh: OIC—Other Interrupt Control Register" on page 396	0000h
3310h	3313h	"Offset 3310h: PRSTS—Power and Reset Status" on page 397	02020000h
3314h	3317h	"Offset 3314h: CIR7—Chipset Initialization Register 7" on page 398	00000000h
3324h	3327h	"Offset 3324h: CIR8—Chipset Initialization Register 8" on page 398	00000000h
3330h	3333h	"Offset 3330h: CIR9—Chipset Initialization Register 9" on page 398	00000000h
3340h	3343h	"Offset 3340h: CIR10—Chipset Initialization Register 10" on page 399	00000000h
3350h	3353h	"Offset 3350h: CIR13—Chipset Initialization Register 13" on page 399	00000000h
3368h	336Bh	"Offset 3368h: CIR14—Chipset Initialization Register 14" on page 399	00000000h
3378h	337Bh	"Offset 3378h: CIR15—Chipset Initialization Register 15" on page 400	00000000h
3388h	338Bh	"Offset 3388h: CIR16—Chipset Initialization Register 16" on page 400	00000000h
33A0h	33A3h	"Offset 33A0h: CIR17—Chipset Initialization Register 17" on page 400	00000000h
33A8h	33ABh	"Offset 33A8h: CIR18—Chipset Initialization Register 18" on page 401	00000000h
33C0h	33C3h	"Offset 33C0h: CIR19—Chipset Initialization Register 19" on page 401	00000000h
33CCh	33CFh	"Offset 33CCh: CIR20—Chipset Initialization Register 20" on page 401	00000000h
33D0h	33D3h	"Offset 33D0h: CIR21—Chipset Initialization Register 21" on page 402	00000000h
33D4h	33D7h	"Offset 33D4h: CIR22—Chipset Initialization Register 22" on page 402	00000000h
3400h	3403h	"Offset 3400h: RC—RTC Configuration Register" on page 403	00000000h

Table 7-1. Chipset Configuration Registers (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
3404h	3407h	"Offset 3404h: HPTC—High Precision Timer Configuration Register" on page 404	00000000h
3410h	3413h	"Offset 3410h: GCS—General Control and Status Register" on page 405	00000yy0h (yy = xx0000x0b)
3414h	3414h	"Offset 3414h: BUC—Backed Up Control Register" on page 407	0000000xb
3418h	341Bh	"Offset 3418h: FD—Function Disable Register" on page 408	See register description
341Ch	341Fh	"Offset 341Ch: CG—Clock Gating" on page 410	00000000h
3420h	3420h	"Offset 3420h: FDSW—Function Disable SUS Well" on page 411	00h
3428h	342Bh	"Offset 3428h: FD2—Function Disable 2" on page 411	00000000h
3500h	3514h	"Offset 3500h: USBIR[0:5]—USB Initialization Register [0-5]" on page 412	20000B4Ah
3564h	3567h	"Offset 3564h: USBIRC—USB Initialization Register C" on page 413	0000371Bh
3570h	3573h	"Offset 3570h: USBIRA—USB Initialization Register A" on page 413	00000000h
357Ch	357Fh	"Offset 357Ch: USBIRB—USB Initialization Register B" on page 414	0001C000h
3590h	3594h	"Offset 3590h: MISCCTL—Miscellaneous Control Register" on page 415	00000000h
359Ch	359Fh	"Offset 359Ch: PDO—USB Port Disable Override" on page 416	0000h
35A0h	35A3h	"Offset 35A0h: USBOCM1—Overcurrent MAP Register 1" on page 417	C0300C03h
35B0h	35B3h	"Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register" on page 418	00000000h

7.1.1.1 Offset 0014h: V0CTL—Virtual Channel 0 Resource Control Register (B0:D31:F0)

Table 7-2. Offset 0014h: V0CTL—Virtual Channel 0 Resource Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0014h **Offset End:** 0017h	
Size: 32 bit	**Default:** 80000023h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	EN	Virtual Channel Enable — Always set to 1. VC0 is always enabled and cannot be disabled.			RO
30 :27	Reserved	Reserved			
26 :24	ID	Virtual Channel Identifier — Indicates the ID to use for this virtual channel.			RO
23 :16	Reserved	Reserved			
15 :08	ETVN	Extended TC/VC Map — Defines the upper 8-bits of the VC0 16-bit TC/VC mapping registers. These registers use the PCI Express* reserved TC[3] traffic class bit.			
07 :01	TVM	Transaction Class / Virtual Channel Map — Indicates which transaction classes are mapped to this virtual channel. When a bit is set, this transaction class is mapped to the virtual channel.			RW
00	Reserved	Reserved			

7.1.1.2 Offset 001Ah: V0STS—Virtual Channel 0 Resource Status Register (B0:D31:F0)

Table 7-3. Offset 001Ah: V0STS—Virtual Channel 0 Resource Status Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 001Ah **Offset End:** 001Bh		
Size: 16 bit	**Default:** 0000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :02	Reserved	Reserved			
01	NP	VC Negotiation Pending — When set, indicates the virtual channel is still being negotiated with ingress ports.			RO
00	ATS	Port Arbitration Tables Status — There is no port arbitration table for this VC, so this bit is reserved at 0.			RO

7.1.1.3 Offset 001Ch: V1CAP—Virtual Channel 1 Resource Capability Register (B0:D31:F0)

Table 7-4. Offset 001Ch: V1CAP—Virtual Channel 1 Resource Capability Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 001Ch **Offset End:** 001Fh		
Size: 32 bit	**Default:** 00008001h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	AT	Port Arbitration Table Offset — Indicates the location of the port arbitration table in the root complex. A value of 0h indicates the table is not present			RO
23	Reserved	Reserved			
22 :16	MTS	Maximum Time Slots — This value is updated by platform BIOS based upon the determination of the number of time slots available in the platform.			RWO
15	RTS	Reject Snoop Transactions — All snoopable transactions on VC1 are rejected.			RO
14	APS	Advanced Packet Switching — This VC is capable of all transactions, not just advanced packet switching transactions.			RO
13 :08	Reserved	Reserved			
07 :00	PAC	Port Arbitration Capability — Indicates the port arbitration capability is hardware-fixed.			

7.1.1.4 Offset 0020h: V1CTL—Virtual Channel 1 Resource Control Register (B0:D31:F0)

Table 7-5. Offset 0020h: V1CTL—Virtual Channel 1 Resource Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0020h **Offset End:** 0023h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	EN	Virtual Channel Enable — Enables the VC when set. Disables the VC when cleared.			RW
30 :28	Reserved	Reserved			
27 :24	ID	Virtual Channel Identifier — Indicates the ID to use for this virtual channel.			RO
23 :16	Reserved	Reserved			
15 :08	ETVN	Extended TC/VC Map — Defines the upper 8-bits of the VC0 16-bit TC/VC mapping registers. These registers use the PCI Express* reserved TC[3] traffic class bit.			
07 :01	TVM	Transaction Class / Virtual Channel Map — Indicates which transaction classes are mapped to this virtual channel. When a bit is set, this transaction class is mapped to the virtual channel.			RW
00	Reserved	Reserved			

7.1.1.5 Offset 0026h: V1STS—Virtual Channel 1 Resource Status Register (B0:D31:F0)

Table 7-6. Offset 0026h: V1STS—Virtual Channel 1 Resource Status Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0026h **Offset End:** 0027h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :02	Reserved	Reserved			
01	NP	VC Negotiation Pending — When set, indicates the virtual channel is still being negotiated with ingress ports.			RO
00	ATS	Port Arbitration Tables Status — Indicates the coherency status of the port arbitration table. This bit is set when LAT (offset 000Ch:bit 0) is written with value 1 and PAS (offset 0014h:bits19:17) has value of 4h. This bit is cleared after the table has been updated.			RO

7.1.1.6 Offset 0050h: CIR0—Chipset Initialization Register 0 (B0:D31:F0)

Table 7-7. Offset 0050h: CIR0—Chipset Initialization Register 0

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0050h **Offset End:** 0053h		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR0 Field 0 — BIOS must set this field.			RW

7.1.1.7 Offset 0088h: CIR1—Chipset Initialization Register 1 (B0:D31:F0)

Table 7-8. Offset 0088h: CIR1—Chipset Initialization Register 1

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0088h **Offset End:** 008Bh		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved			
20		CIR1 Field 3 — BIOS must set this bit.			RWO
19 :16	Reserved	Reserved			
15		CIR1 Field 2 — BIOS must set this bit.			RWO
14 :13	Reserved	Reserved			
12		CIR1 Field 1 — BIOS must set this bit.			RWO
11 :00	Reserved	Reserved			

7.1.1.8 Offset 00ACh: REC—Root Error Command Register (B0:D31:F0)

Table 7-9. Offset 00ACh: REC—Root Error Command Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 00ACh **Offset End:** 00AFh	
Size: 32 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	DPDP	Drop Poisoned Downstream Packets — Determines how downstream packets on DMI are handled that are received with the EP field set, indicating poisoned data: 1 = This packet and all subsequent packets with data received on DMI for any VC will have their Unsupported Transaction (UT) field set causing them to master Abort downstream. Packets without data such as memory, IO and config read requests are allowed to proceed. 0 = Packets are forwarded downstream without forcing the UT field set.			RW
30 :00	Reserved	Reserved			

7.1.1.9 Offset 01A0h: ILCL—Internal Link Capabilities List Register (B0:D31:F0)

Table 7-10. Offset 01A0h: ILCL—Internal Link Capabilities List Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 01A0h **Offset End:** 01A3h	
Size: 32 bit	**Default:** 00010006h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	NEXT	Next Capability Offset — Indicates this is the last item in the list.			RO
19 :16	CV	Capability Version — Indicates the version of the capability structure.			RO
15 :00	CID	Capability ID — Indicates this is capability for DMI.			RO

7.1.1.10 Offset 01A4h: LCAP—Link Capabilities Register (B0:D31:F0)

Table 7-11. Offset 01A4h: LCAP—Link Capabilities Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 01A4h **Offset End:** 01A7h	
Size: 32 bit	**Default:** 00012841h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved			
17 :15	EL1	L1 Exit Latency — L1 is supported on DMI.			RO
14 :12	EL0	L0s Exit Latency — This field indicates that exit latency is 128 ns to less than 256 ns.			RWO
11 :10	APMS	Active State Link PM Support — Indicates that L0s and L1 are supported on DMI.			RWO
09 :04	MLW	Maximum Link Width — Indicates the maximum link width is 4 ports.			RO
03 :00	MLS	Maximum Link Speed — Indicates the link speed is 2.5 Gb/s.			RO

7.1.1.11 Offset 01A8h: LCTL—Link Control Register (B0:D31:F0)

Table 7-12. Offset 01A8h: LCTL—Link Control Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 01A8h **Offset End:** 01A9h	
Size: 32 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	Reserved	Reserved			
07	ES	Extended Synch — When set, forces extended transmission of FTS ordered sets when exiting L0s prior to entering L0.			RW
06 :02	Reserved	Reserved			
01 :00	ASPM	Active State Link PM Control — Indicates whether DMI should enter L0s. 00 = Disabled 01 = L0s entry enabled 10 = Reserved 11 = Reserved			RW

7.1.1.12 Offset 01AAh: LSTS—Link Status Register (B0:D31:F0)

Table 7-13. Offset 01AAh: LSTS—Link Status Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 01AAh **Offset End:** 01ABh	
Size: 16 bit	**Default:** 0041h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09 :04	NLW	Negotiated Link Width — Negotiated link width is x4 (000100b).			RO
03 :00	LS	Link Speed — Link is 2.5 Gb/s.			RO

7.1.1.13 Offset 0220h: BCR—Backbone Configuration Register (B0:D31:F0)

Table 7-14. Offset 0220h: BCR—Backbone Configuration Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0220h **Offset End:** 0223h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	Reserved	Reserved			
06		BCR Field 2 — BIOS must set this bit.			RW
05 :03	Reserved	Reserved			
02 :00		BCR Field 1 — BIOS program this field to 101b			RW

7.1.1.14 Offset 0224h: RPC—Root Port Configuration Register (B0:D31:F0)

Table 7-15. Offset 0224h: RPC—Root Port Configuration Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0224h **Offset End:** 0227h	
Size: 32 bit	**Default:** 0000000yh (y = 00xxb)			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07	HPE	High Priority Port Enable: 0 = The high priority path is not enabled. 1 = The port selected by the HPP field in this register is enabled for high priority. It will be arbitrated above all other VC0 (including integrated VC0) devices.			RW
06 :04	HPP	High Priority Port — This controls which port is enabled for high priority when the HPE bit in this register is set. 101 = Port 4 010 = Port 3 001 = Port 2 000 = Port 1			RW
03 :02	Reserved	Reserved			
01 :00	PC	Port Configuration — This controls how the PCI bridges are organized in various modes of operation for Ports 1-4. For the following mappings, if a port is not shown, it is considered a x1 port with no connection. These bits are set by the PCIEPCS1[1:0] soft strap. 11 = 1 x4, Port 1 (x4) 10 = 2x2, Port 1 (x2), Port 3 (x2) 01 = 1x2 and 2x1s, Port 1 (x2), Port 3 (x1) and Port 4 (x1) 00 = 4 x1s, Port 1 (x1), Port 2 (x1), Port 3 (x1) and Port 4 (x1) These bits are in the resume well and are only reset by RSMRST#.			RO

7.1.1.15 Offset 0234h: DMIC—DMI Control Register (B0:D31:F0)

Table 7-16. Offset 0234h: DMIC—DMI Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0234h **Offset End:** 0237h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved			
01 :00	DMICGEN	DMI Clock Gate Enable — BIOS must program this field to 11b.			RW

7.1.1.16 Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports (B0:D31:F0)

For the PCI Express* root ports, the assignment of a function number to a root port is not fixed. BIOS may re-assign the function numbers on a port by port basis. This capability will allow BIOS to disable/hide any root port and still have functions 0 thru N-1 where N is the total number of enabled root ports.

Port numbers will remain fixed to a physical root port.

The existing root port Function Disable registers operate on physical ports (not functions).

Port Configuration (1x4, 4x1, etc.) is not affected by the logical function number assignment and is associated with physical ports.

Table 7-17. Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 0238h **Offset End:** 023Ch	
Size: 32 bit	**Default:** 76543210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	RP4CH	Root Port 4 Config Hide — This bit is used to hide the root port and any devices behind it from being discovered by the OS. When set to '1' the root port will not claim any downstream configuration transactions.			RWO
14 :12	RP4FN	Root Port 4 Function Number — These bits set the function number for PCI Express* Root Port 4. This root port function number must be a unique value from the other root port function number.			RW

Table 7-17. Offset 0238h: RPFN—Root Port Function Number and Hide for PCI Express* Root Ports

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0238h **Offset End:** 023Ch	
Size: 32 bit	**Default:** 76543210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	RP4CH	Root Port 3 Config Hide — This bit is used to hide the root port and any devices behind it from being discovered by the OS. When set to '1' the root port will not claim any downstream configuration transactions.			RWO
10 :08	RP3FN	Root Port 3 Function Number — These bits set the function number for PCI Express* Root Port 3. This root port function number must be a unique value from the other root port function number.			RW
07	RP3CH	Root Port 2 Config Hide — This bit is used to hide the root port and any devices behind it from being discovered by the OS. When set to '1' the root port will not claim any downstream configuration transactions.			RWO
06 :04	RP2FN	Root Port 2 Function Number — These bits set the function number for PCI Express* Root Port 2. This root port function number must be a unique value from the other root port function number.			RW
03	RP2CH	Root Port 1 Config Hide — This bit is used to hide the root port and any devices behind it from being discovered by the OS. When set to '1' the root port will not claim any downstream configuration transactions.			RWO
02 :00	RP1FN	Root Port 1 Function Number — These bits set the function number for PCI Express* Root Port 1. This root port function number must be a unique value from the other root port function number.			RW

7.1.1.17 Offset 0290h: Reserved Register (B0:D31:F0)

Table 7-18. Offset 0290h: Reserved

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 0290h **Offset End:** 0293h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Reserved	Reserved			

7.1.1.18 Offset 1D40H: CIR5—Chipset Initialization Register 5 (B0:D31:F0)

Table 7-19. Offset 1D40H: CIR5—Chipset Initialization Register 5

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 1D40h **Offset End:** 1D47h	
Size: 64 bit	**Default:** 00000000000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :01	Reserved	Reserved			
00		CIR5 Field 1 — BIOS must program this field to 1b.			RW

7.1.1.19 Offset 1E00h: TRSR—Trap Status Register (B0:D31:F0)

Table 7-20. Offset 1E00h: TRSR—Trap Status Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 1E00h **Offset End:** 1E03h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03 :00	CTSS	Cycle Trap SMI# Status — These bits are set by hardware when the corresponding Cycle Trap register is enabled and a matching cycle is received (and trapped). These bits are OR'ed together to create a single status bit in the Power Management register space. The SMI# and trapping must be enabled in order to set these bits. These bits are set before the completion is generated for the trapped cycle, thereby ensuring that the processor can enter the SMI# handler when the instruction completes. Each status bit is cleared by writing a 1 to the corresponding bit location in this register.			RWC

7.1.1.20 Offset 1E10h: TRCR—Trapped Cycle Register (B0:D31:F0)

This register saves information about the I/O Cycle that was trapped and generated the SMI# for software to read.

Table 7-21. Offset 1E10h: TRCR—Trapped Cycle Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 1E10h **Offset End:** 1E17h		
Size: 64 bit	**Default:** 0000000000000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :25	Reserved	Reserved			
24	RWI	Read/Write#: 0 = Trapped cycle was a write cycle. 1 = Trapped cycle was a read cycle.			RO
23 :20	Reserved	Reserved			
19 :16	AHBE	Active-high Byte Enables — This is the dword-aligned byte enables associated with the trapped cycle. A 1 in any bit location indicates that the corresponding byte is enabled in the cycle.			RO
15 :02	TIOA	Trapped I/O Address — This is the dword-aligned address of the trapped cycle.			RO
01 :00	Reserved	Reserved			

7.1.1.21 Offset 1E18h: TWDR—Trapped Write Data Register (B0:D31:F0)

This register saves the data from I/O write cycles that are trapped for software to read.

Table 7-22. Offset 1E18h: TWDR—Trapped Write Data Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 1E18h **Offset End:** 1E1Fh		
Size: 64 bit	**Default:** 0000000000000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :32	Reserved	Reserved			
31 :00	TIOD	Trapped I/O Data — Dword of I/O write data. This field is undefined after trapping a read cycle.			RO

7.1.1.22 Offset 1E80h: IOTRn—I/O Trap Register (0-3) (B0:D31:F0)

Offset Address:

1E80h-1E87h Register 0
1E88h-1E8Fh Register 1
1E90h-1E97h Register 2
1E98h-1E9Fh Register 3

These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Table 7-23. Offset 1E80h: IOTRn—I/O Trap Register (0-3)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 1E80h **Offset End:** 1E9Fh	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :50	Reserved	Reserved			
49	RWM	Read/Write Mask: 0 = The cycle must match the type specified in bit 48. 1 = Trapping logic will operate on both read and write cycles.			RW
48	RWIO	Read/Write#: 0 = Write 1 = Read The value in this field does not matter if bit 49 is set.			RW
47 :40	Reserved	Reserved			
39 :36	BEM	Byte Enable Mask — A 1 in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.			RW
35 :32	TBE	Byte Enables (TBE) — R/W. Active-high dword-aligned byte enables.			RW
31 :24	Reserved	Reserved			
23 :18	ADMA	Address[7:2] Mask — A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the dword address, allowing for traps on address ranges up to 256 bytes in size.			RW
17 :16	Reserved	Reserved			
15 :02	IOAD	I/O Address[15:2] — Dword-aligned address			RW
01	Reserved	Reserved			
00	TRSE	Trap and SMI# Enable: 0 = Trapping and SMI# logic disabled. 1 = The trapping logic specified in this register is enabled.			RW

7.1.1.23 Offset 2010h: DMC—DMI Miscellaneous Control Register (B0:D31:F0)

Table 7-24. Offset 2010h: DMC—DMI Miscellaneous Control Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2010h **Offset End:** 2013h	
Size: 32 bit	**Default:** 00000002h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved			
19	DMC	DMI Misc. Control Field 1 — BIOS shall always program this field as per the BIOS Specification. 0 = Disable DMI Power Savings. 1 = Enable DMI Power Savings.			RW
18 :00	Reserved	Reserved			

7.1.1.24 Offset 2024h: CIR6—Chipset Initialization Register 6 (B0:D31:F0)

Table 7-25. Offset 2024h: CIR6—Chipset Initialization Register 6

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2024h **Offset End:** 2027h	
Size: 32 bit	**Default:** 0B4030C0h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :21		CIR6 Field 2 — BIOS must program this field to 011b.			RW
20 :08	Reserved	Reserved			
31 :08	Reserved	Reserved			
07		CIR6 Field 1 — BIOS must clear this bit.			RW
06 :00	Reserved	Reserved			

7.1.1.25 Offset 2324h: DMC2—DMI Miscellaneous Control Register 2 (B0:D31:F0)

Table 7-26. Offset 2324h: DMC2—DMI Miscellaneous Control Register 2

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2324h **Offset End:** 2327h	
Size: 32 bit	**Default:** 0FFF0FFFh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
27 :16	DMC2	DMI Misc. Control Field 2 — BIOS shall always program this field as per the BIOS Specification.			RW
15 :00	Reserved	Reserved			

7.1.1.26 Offset 60h: SBI Unified AFE Address Register

Table 7-27. Offset 60h: SBI Unified AFE Address Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 2330h **Offset End:** 2333h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :00	OFFSET	IAddress Offset: Register address offset. The contents of this register are sent in the IOSF Sideband Message Register Access address[15:0] field. See the MPHY chapters for register address offsets.		0000h	RW

7.1.1.27 Offset 64h: SSBI Unified AFE Data Register

Table 7-28. Offset 64h: SBI Unified AFE Data Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 2334h **Offset End:** 2337h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DATA	AFE Data: Reads and Writes to this register will trigger an IOSF Sideband access on the private Unified AFE sideband interface. All MPHY accesses are sent as IOSF non-posted transactions. Software needs to ensure the MPHY interface is not busy NOT(MPHYSTAT.BUSY) before writing to this register. A read to this register will trigger a read of the designated MPHY register on the private IOSF sideband interface and when a completion is returned on the sideband interface, a completion can be returned on the IOSF primary interface. Implementation Note: The DMI registers are in memory mapped space. Posted writes to this register will cause a non-posted transaction to occur on the MPHY private sideband interface. The register block must not block accepting of posted writes to the general DMI register space while waiting for a non-posted transaction to complete on the IOSF sideband interface. If software erroneously issues multiple posted writes to this register while MPHYSTAT.BUSY is '1', then the register block is allowed to block accepting of subsequent posted transactions while waiting for the IOSF sideband transaction to complete.			RW

7.1.1.28 Offset 68h: SBI Unified AFE Status Register

Table 7-29. Offset 68h: SBI Unified AFE Status Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 2338h **Offset End:** 2339h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :03	Reserved	Reserved			
02 :01	RESP	Response Status (RESPONSE): 00 – Successful 01 – Unsuccessful / Not Supported 10 – Reserved 11 – Powered Down This register reflects the response status for the previously completed transaction. The value of this register is only meaningful if NOT(MPHYSTAT.BUSY).		00b	RO
00	BUSY	Busy / Ready#: '0': The MPHY IOSF sideband interface is ready for a new transaction '1': The MPHY IOSF sideband interface is busy with the previous transaction.		0b	RO

7.1.1.29 Offset 3000h: TCTL—TCO Configuration Register (B0:D31:F0)

Table 7-30. Offset 3000h: TCTL—TCO Configuration Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3000h **Offset End:** 3000h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	IE	TCO IRQ Enable: 0 = TCO IRQ is disabled. 1 = TCO IRQ is enabled, as selected by the TCO_IRQ_SEL field.			RW
06 :03	Reserved	Reserved			
02 :00	IS	TCO IRQ Select — Specifies on which IRQ the TCO will internally appear. If not using the APIC, the TCO interrupt must be routed to IRQ9-11, and that interrupt is not sharable with the SERIRQ stream, but is shareable with other PCI interrupts. If using the APIC, the TCO interrupt can also be mapped to IRQ20-23, and can be shared with other interrupt. 000 = IRQ 9 001 = IRQ 10 010 = IRQ 11 011 = Reserved 100 = IRQ 20 (only if APIC enabled) 101 = IRQ 21 (only if APIC enabled) 110 = IRQ 22 (only if APIC enabled) 111 = IRQ 23 (only if APIC enabled) When setting the these bits, the IE bit should be cleared to prevent glitching. When the interrupt is mapped to APIC interrupts 9, 10 or 11, the APIC should be programmed for active-high reception. When the interrupt is mapped to APIC interrupts 20 through 23, the APIC should be programmed for active-low reception.			RW

7.1.1.30 Offset 3100h: D31IP—Device 31 Interrupt Pin Register (B0:D31:F0)

Table 7-31. Offset 3100h: D31IP—Device 31 Interrupt Pin Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3100h **Offset End:** 3103h	
Size: 32 bit	**Default:** 03243200h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
27 :24	TSIP	Thermal Sensor Pin — Indicates which pin the Thermal Sensor controller drives as its interrupt 0h = No interrupt 1h = INTA# 2h = INTB# (Default) 3h = INTC# 4h = INTD# 5h–Fh = Reserved			RW
23 :20	SIP2	SATA* Pin 2 — Indicates which pin the SATA* controller 2 drives as its interrupt. 0h = No interrupt 1h = INTA# 2h = INTB# (Default) 3h = INTC# 4h = INTD# 5h–Fh = Reserved			RW
19 :16	Reserved	Reserved			
15 :12	SMIP	SMBus Pin — Indicates which pin the SMBus controller drives as its interrupt. 0h = No interrupt 1h = INTA# 2h = INTB# (Default) 3h = INTC# 4h = INTD# 5h–Fh = Reserved			RW
11 :08	SIP	SATA* Pin — Indicates which pin the SATA* controller drives as its interrupt. 0h = No interrupt 1h = INTA# 2h = INTB# (Default) 3h = INTC# 4h = INTD# 5h–Fh = Reserved			RW
07 :04	Reserved	Reserved			
03 :00	LIP	LPC Bridge Pin — Currently, the LPC bridge does not generate an interrupt, so this field is read-only and 0.			RO

7.1.1.31 Offset 3108h: D29IP—Device 29 Interrupt Pin Register (B0:D31:F0)

Table 7-32. Offset 3108h: D29IP—Device 29 Interrupt Pin Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3108h **Offset End:** 310Bh	
Size: 32 bit	**Default:** 10004321h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03 :00	E1P	EHCI #1 — Indicates which interrupt the EHCI controller #1 drives as its interrupt. 0h = No interrupt 1h = INTA (Default) 2h = INTB 3h = INTC 4h = INTD 5h-7h = Reserved Note: EHCI Controller #1 is mapped to Device 29 Function 0 when RMH is enabled			RW

7.1.1.32 Offset 310Ch: D28IP—Device 28 Interrupt Pin Register (B0:D31:F0)

Table 7-33. Offset 310Ch: D28IP—Device 28 Interrupt Pin Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 310Ch **Offset End:** 310Fh	
Size: 32 bit	**Default:** 00214321h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :12	P4IP	PCI Express* #4 Pin — Indicates which pin PCI Express* port #4 drives as its interrupt. 0h = No interrupt 1h = INTA 2h = INTB 3h = INTC 4h = INTD (Default) 5h-7h = Reserved			RW
11 :08	P3IP	PCI Express* #3 Pin — Indicates which pin PCI Express* port #3 drives as its interrupts. 0h = No interrupt 1h = INTA 2h = INTB 3h = INTC (Default) 4h = INTD 5h-7h = Reserved			RW
07 :04	P2IP	PCI Express* #2 Pin — Indicates which pin PCI Express* port #2 drives as its interrupts. 0h = No interrupt 1h = INTA 2h = INTB (Default) 3h = INTC 4h = INTD 5h-7h = Reserved			RW
03 :00	P1IP	PCI Express* #1 Pin — Indicates which pin PCI Express* port #1 drives as its interrupt. 0h = No interrupt 1h = INTA (Default) 2h = INTB 3h = INTC 4h = INTD 5h-7h = Reserved			RW

7.1.1.33 Offset 3124h: D22IP—Device 22 Interrupt Pin Register (B0:D31:F0)

Table 7-34. Offset 3124h: D22IP—Device 22 Interrupt Pin Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3124h **Offset End:** 3127h	
Size: 32 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :12	KTIP	KT Pin — Indicates which pin the Keyboard text PCI functionality drives as its interrupt. 0h = No Interrupt 1h = INTA 2h = INTB 3h = INTC 4h = INTD 5h-Fh = Reserved			RW
11 :08	IDERIP	IDER Pin — Indicates which pin the IDE Redirect PCI functionality drives as its interrupt. 0h = No Interrupt 1h = INTA 2h = INTB 3h = INTC 4h = INTD 5h-Fh = Reserved			RW
07 :04	MEI2IP	Intel(R) MEI #2 Pin — Indicates which pin the Management Engine Interface Controller #2 drives as its interrupt. 0h = No Interrupt 1h = INTA 2h = INTB 3h = INTC 4h = INTD 5h-Fh = Reserved			RW
03 :00	MEI1IP	Intel(R) MEI #1 Pin — Indicates which pin the Management Engine Interface Controller #1 drives as its interrupt 0h = No Interrupt 1h = INTA 2h = INTB 3h = INTC 4h = INTD 5h-Fh = Reserved			RW

7.1.1.34 Offset 3140h: D31IR—Device 31 Interrupt Route Register (B0:D31:F0)

Table 7-35. Offset 3140h: D31IR—Device 31 Interrupt Route Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 3140h **Offset End:** 3141h	
Size: 16 bit	**Default:** 3210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			O
14 :12	IDR	Interrupt D Pin Route — Indicates which physical pin on the PCH is connected to the INTD# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# (Default) 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
11	Reserved	Reserved			O
10 :08	ICR	Interrupt C Pin Route — Indicates which physical pin on the PCH is connected to the INTC# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# (Default) 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
07	Reserved	Reserved			O
06 :04	IBR	Interrupt B Pin Route — Indicates which physical pin on the PCH is connected to the INTB# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# (Default) 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
03	Reserved	Reserved			O
02 :00	IAR	Interrupt A Pin Route — Indicates which physical pin on the PCH is connected to the INTA# pin reported for device 31 functions. 0h = PIRQA# (Default) 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW

7.1.1.35 Offset 3144h: D29IR—Device 29 Interrupt Route Register (B0:D31:F0)

Table 7-36. Offset 3144h: D29IR—Device 29 Interrupt Route Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3144h **Offset End:** 3145h	
Size: 16 bit	**Default:** 3210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			O
14 :12	IDR	Interrupt D Pin Route — Indicates which physical pin on the PCH is connected to the INTD# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# (Default) 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
11	Reserved	Reserved			O
10 :08	ICR	Interrupt C Pin Route — Indicates which physical pin on the PCH is connected to the INTC# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# (Default) 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
07	Reserved	Reserved			O
06 :04	IBR	Interrupt B Pin Route — Indicates which physical pin on the PCH is connected to the INTB# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# (Default) 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
03	Reserved	Reserved			O
02 :00	IAR	Interrupt A Pin Route — Indicates which physical pin on the PCH is connected to the INTA# pin reported for device 31 functions. 0h = PIRQA# (Default) 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW

7.1.1.36 Offset 3146h: D28IR—Device 28 Interrupt Route Register (B0:D31:F0)

Table 7-37. Offset 3146h: D28IR—Device 28 Interrupt Route Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3146h **Offset End:** 3147h	
Size: 16 bit	**Default:** 3210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			O
14 :12	IDR	Interrupt D Pin Route — Indicates which physical pin on the PCH is connected to the INTD# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# (Default) 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
11	Reserved	Reserved			O
10 :08	ICR	Interrupt C Pin Route — Indicates which physical pin on the PCH is connected to the INTC# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# (Default) 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
07	Reserved	Reserved			O
06 :04	IBR	Interrupt B Pin Route — Indicates which physical pin on the PCH is connected to the INTB# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# (Default) 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
03	Reserved	Reserved			O
02 :00	IAR	Interrupt A Pin Route — Indicates which physical pin on the PCH is connected to the INTA# pin reported for device 31 functions. 0h = PIRQA# (Default) 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW

7.1.1.37 Offset 315Ch: D22IR—Device 22 Interrupt Route Register (B0:D31:F0)

Table 7-38. Offset 315Ch: D22IR—Device 22 Interrupt Route Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 315Ch **Offset End:** 315Dh	
Size: 16 bit	**Default:** 3210h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			O
14 :12	IDR	Interrupt D Pin Route — Indicates which physical pin on the PCH is connected to the INTD# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# (Default) 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
11	Reserved	Reserved			O
10 :08	ICR	Interrupt C Pin Route — Indicates which physical pin on the PCH is connected to the INTC# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# 2h = PIRQC# (Default) 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
07	Reserved	Reserved			O
06 :04	IBR	Interrupt B Pin Route — Indicates which physical pin on the PCH is connected to the INTB# pin reported for device 31 functions. 0h = PIRQA# 1h = PIRQB# (Default) 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW
03	Reserved	Reserved			O
02 :00	IAR	Interrupt A Pin Route — Indicates which physical pin on the PCH is connected to the INTA# pin reported for device 31 functions. 0h = PIRQA# (Default) 1h = PIRQB# 2h = PIRQC# 3h = PIRQD# 4h = PIRQE# 5h = PIRQF# 6h = PIRQG# 7h = PIRQH#			RW

7.1.1.38 Offset 31FEh: OIC—Other Interrupt Control Register (B0:D31:F0)

FEC10000h - FEC3FFFFh is allocated to PCIe* when I/OxApic Enable (PAE) bit is set.

Table 7-39. Offset 31FEh: OIC—Other Interrupt Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 31FEh **Offset End:** 31FFh		
Size: 16 bit	**Default:** 0000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved			
11	TPMINTPOL	Intel® TPM Interrupt Polarity Enable: When this bit is cleared to 0 the Intel TPM IRQ signal will be forced to operate in ActiveHigh mode regardless of the setting of the IOAPIC RTE for the corresponding interrupt. When this bit is set to 1 the Intel TPM IRQ signal's polarity will be set to match the corresponding IOAPIC RTE polarity. If the IOAPIC is programmed for ActiveLow mode the corresponding Intel TPM IRQ will operate in ActiveLow mode. If the IOAPIC RTE is programmed for ActiveHigh mode the corresponding Intel TPM IRQ will operate in ActiveHigh mode.			
10	Reserved	Reserved			
09	CEN	Coprocessor Error Enable: 0 = FERR# will not generate IRQ13 nor IGNNE#. I1 = f FERR# is low, the PCH generates IRQ13 internally and holds it until an I/O port F0h write. It will also drive IGNNE# active.			RW
08	AEN	APIC Enable: 0 = The internal IOxAPIC is disabled. 1 = Enables the internal IOxAPIC and its address decode. Note: SW should read this register after modifying APIC enable bit prior to access to the IOxAPIC address range.			RW
07 :00	ASEL	APIC Range Select — These bits define address bits 19:12 for the IOxAPIC range. The default value of 00h enables compatibility with prior PCH products as an initial value. This value must not be changed unless the IOxAPIC Enable bit is cleared.			RW

7.1.1.39 Offset 3310h: PRSTS—Power and Reset Status (B0:D31:F0)

Table 7-40. Offset 3310h: PRSTS—Power and Reset Status

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function :** B0:D31 :F0		**Offset Start:** 3310h **Offset End:** 3313h	
Size: 32 bit	**Default:** 02020000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15		Power Management Watchdog Timer — This bit is set when the Power Management watchdog timer causes a global reset.			RWC
14 :08	Reserved	Reserved			
07		VE Watchdog Timer Status — This bit is set when the VE watchdog timer causes a global reset.			RWC
06		Intel® Management Engine Watchdog Timer Status — This bit is set when the Intel® Management Engine watchdog timer causes a global reset.			RWC
05 :03	Reserved	Reserved			
02	ME_HRST_WARM_STS	ME Host Reset Warm Status — This bit is set when the Intel® Management Engine generates a Host reset without power cycling. Software clears this bit by writing a 1 to this bit position.			RWC
01	ME_HRST_COLD_STS	ME Host Reset Cold Status — This bit is set when the Intel® Management Engine generates a Host reset with power cycling. Software clears this bit by writing a 1 to this bit position.			RWC
00	ME_WAKE_STS	ME WAKE STATUS — This bit is set when the Intel® Management Engine generates a Non-Maskable wake event, and is not affected by any other enable bit. When this bit is set, the Host Power Management logic wakes to S0.			RWC

7.1.1.40 Offset 3314h: CIR7—Chipset Initialization Register 7 (B0:D31:F0)

Table 7-41. Offset 3314h: CIR7—Chipset Initialization Register 7

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3314h **Offset End:** 3317h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03 :00		CIR7 Field 1— BIOS must program this field to 1111b.			RW

7.1.1.41 Offset 3324h: CIR8—Chipset Initialization Register 8 (B0:D31:F0)

Table 7-42. Offset 3324h: CIR8—Chipset Initialization Register 8

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3324h **Offset End:** 3327h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR8 Field 1 — BIOS must program this field to 04000000h.			RW

7.1.1.42 Offset 3330h: CIR9—Chipset Initialization Register 9 (B0:D31:F0)

Table 7-43. Offset 3330h: CIR9—Chipset Initialization Register 9

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3330h **Offset End:** 3333h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR9 Field 1 — BIOS must program this field to 00000000h.			RW

7.1.1.43 Offset 3340h: CIR10—Chipset Initialization Register 10 (B0:D31:F0)

Table 7-44. Offset 3340h: CIR10—Chipset Initialization Register 10

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3340h **Offset End:** 3343h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR10 Field 1 — BIOS must program this field.			RW

7.1.1.44 Offset 3350h: CIR13—Chipset Initialization Register 13 (B0:D31:F0)

Table 7-45. Offset 3350h: CIR13—Chipset Initialization Register 13

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3350h **Offset End:** 3353h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR13 Field 1 — BIOS must program this field.			RW

7.1.1.45 Offset 3368h: CIR14—Chipset Initialization Register 14 (B0:D31:F0)

Table 7-46. Offset 3368h: CIR14—Chipset Initialization Register 14

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3368h **Offset End:** 336Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR14 Field 1 — BIOS must program this field.			RW

7.1.1.46 Offset 3378h: CIR15—Chipset Initialization Register 15 (B0:D31:F0)

Table 7-47. Offset 3378h: CIR15—Chipset Initialization Register 15

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3378h **Offset End:** 337Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR15 Field 1 — BIOS must program this field.			RW

7.1.1.47 Offset 3388h: CIR16—Chipset Initialization Register 16 (B0:D31:F0)

Table 7-48. Offset 3388h: CIR16—Chipset Initialization Register 16

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3388h **Offset End:** 338Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR16 Field 1 — BIOS must program this field.			RW

7.1.1.48 Offset 33A0h: CIR17—Chipset Initialization Register 17 (B0:D31:F0)

Table 7-49. Offset 33A0h: CIR17—Chipset Initialization Register 17

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 33A0h **Offset End:** 33A3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR17 Field 1 — BIOS must program this field.			RW

7.1.1.49 Offset 33A8h: CIR18—Chipset Initialization Register 18 (B0:D31:F0)

Table 7-50. Offset 33A8h: CIR18—Chipset Initialization Register 18

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 33A8h **Offset End:** 33ABh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR18 Field 1 — BIOS must program this field.			RW

7.1.1.50 Offset 33C0h: CIR19—Chipset Initialization Register 19 (B0:D31:F0)

Table 7-51. Offset 33C0h: CIR19—Chipset Initialization Register 19

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 33C0h **Offset End:** 33C3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR19 Field 1 — BIOS must program this field.			RW

7.1.1.51 Offset 33CCh: CIR20—Chipset Initialization Register 20 (B0:D31:F0)

Table 7-52. Offset 33CCh: CIR20—Chipset Initialization Register 20

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 33CCh **Offset End:** 33CFh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR20 Field 1 — BIOS must program this field.			RW

7.1.1.52 Offset 33D0h: CIR21—Chipset Initialization Register 21 (B0:D31:F0)

Table 7-53. Offset 33D0h: CIR21—Chipset Initialization Register 21

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 33D0h **Offset End:** 33D3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR21 Field 1 — BIOS must program this field.			RW

7.1.1.53 Offset 33D4h: CIR22—Chipset Initialization Register 22 (B0:D31:F0)

Table 7-54. Offset 33D4h: CIR22—Chipset Initialization Register 22

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 33D4h **Offset End:** 33D7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CIR22 Field 1 — BIOS must program this field. Program this register after all registers in the 3330-33D3 range and B0:D31:F0:A9h are already programmed			RW

7.1.1.54 Offset 3400h: RC—RTC Configuration Register (B0:D31:F0)

Table 7-55. Offset 3400h: RC—RTC Configuration Register

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3400h **Offset End:** 3403h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04	UL	Upper 128 Byte Lock: 0 = Bytes not locked. 1 = Bytes 38h-3Fh in the upper 128-byte bank of RTC RAM are locked and cannot be accessed. Writes will be dropped and reads will not return any ensured data. Bit reset on system reset.			RWLO
03	LL	Lower 128 Byte Lock: 0 = Bytes not locked. 1 = Bytes 38h-3Fh in the lower 128-byte bank of RTC RAM are locked and cannot be accessed. Writes will be dropped and reads will not return any ensured data. Bit reset on system reset.			RWLO
02	UE	Upper 128 Byte Enable: 0 = Bytes locked. 1 = The upper 128-byte bank of RTC RAM can be accessed.			RW
01 :00	Reserved	Reserved			

7.1.1.55 Offset 3404h: HPTC—High Precision Timer Configuration Register (B0:D31:F0)

Table 7-56. Offset 3404h: HPTC—High Precision Timer Configuration Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3404h **Offset End:** 3407h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07	AE	Address Enable: 0 = Address disabled. 1 = The PCH will decode the High Precision Timer memory address range selected by bits 1:0 below.			RW
06 :02	Reserved	Reserved			
01 :00	AS	Address Select — This 2-bit field selects 1 of 4 possible memory address ranges for the High Precision Timer functionality. The encodings are: 00 = FED0_0000h – FED0_03FFh 01 = FED0_1000h – FED0_13FFh 10 = FED0_2000h – FED0_23FFh 11 = FED0_3000h – FED0_33FFh			RW

7.1.1.56 Offset 3410h: GCS—General Control and Status Register (B0:D31:F0)

Table 7-57. Offset 3410h: GCS—General Control and Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3410h **Offset End:** 3413h	
Size: 32 bit	**Default:** 00000yy0h (yy = xx0000x0b)			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :13	Reserved	Reserved			
12	FLRCSSEL	Function Level Reset Capability Structure Select: 0 = Function Level Reset (FLR) will utilize the standard capability structure with unique capability ID assigned by PCISIG. 1 = Vendor Specific Capability Structure is selected for FLR.			RW
11 :10	BBS	Boot BIOS Straps — This field determines the destination of accesses to the BIOS memory range. The default values for these bits represent the strap values of BBS1# /GPIO51 (bit 11) at the rising edge of PWROK and BBS0# (bit 10) at the rising edge of MEPWROK. **Bits 11:10** / **Description** 00b — RESERVED 01b — RESERVED 10b — RESERVED 11b — SPI When SPI is selected, the range that is decoded is further qualified by other configuration bits described in the respective sections. The value in this field can be overwritten by software as long as the BIOS Interface Lock-Down (bit 0) is not set. Note: Booting to PCI is intended for debug/testing only. Boot BIOS Destination Select to LPC/PCI by functional strap or via Boot BIOS Destination Bit will not affect SPI accesses initiated by Intel® Management Engine or Integrated GbE LAN.			RW
09	SERM	Server Error Reporting Mode: 0 = The PCH is the final target of all errors. The processor sends a messages to the PCH for the purpose of generating NMI. 1 = The processor is the final target of all errors from PCI Express* and DMI. In this mode, if the PCH detects a fatal, non-fatal, or correctable error on DMI or its downstream ports, it sends a message to the Processor. If the PCH receives an ERR_* message from the downstream port, it sends that message to the Processor.			RW
08 :07	Reserved	Reserved			
06	FME	FERR# MUX Enable — This bit enables FERR# to be a processor break event indication. 0 = Disabled. 1 = The PCH examines FERR# during a C2, C3, or C4 state as a break event.			RW

Table 7-57. Offset 3410h: GCS—General Control and Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3410h **Offset End:** 3413h	
Size: 32 bit	**Default:** 00000yy0h (yy = xx0000x0b)			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05	NR	No Reboot — This bit is set when the "No Reboot" strap (NRBOOTS pin on the PCH) is sampled high on PWROK. This bit may be set or cleared by software if the strap is sampled low but may not override the strap when it indicates "No Reboot". 0 = System will reboot upon the second timeout of the TCO timer. 1 = The TCO timer will count down and generate the SMI# on the first timeout, but will not reboot on the second timeout.			RW
04	AME	Alternate Access Mode Enable: 0 = Disabled. 1 = Alternate access read only registers can be written, and write only registers can be read. Before entering a low power state, several registers from powered down parts may need to be saved. In the majority of cases, this is not an issue, as registers have read and write paths. However, several of the ISA compatible registers are either read only or write only. To get data out of write-only registers, and to restore data into read-only registers, the PCH implements an alternate access mode.			RW
03	SPS	Shutdown Policy Select — When cleared (default), the PCH will drive INIT# in response to the shutdown Vendor Defined Message (VDM). When set to 1, the PCH will treat the shutdown VDM similar to receiving a CF9h I/O write with data value06h, and will drive PLTRST# active.			RW
02	RPR	Reserved Page Route — Determines where to send the reserved page registers. These addresses are sent to PCI or LPC for the purpose of generating POST codes. The I/O addresses modified by this field are: 80h, 84h, 85h, 86h, 88h, 8Ch, 8Dh, and 8Eh. 0 = Writes will be forwarded to LPC, shadowed within the PCH, and reads will be returned from the internal shadow 1 = Reserved The aliases for these registers, at 90h, 94h, 95h, 96h, 98h, 9Ch, 9Dh, and 9Eh, are always decoded to LPC.			RW
01	Reserved	Reserved			
00	BILD	BIOS Interface Lock-Down: 0 = Disabled. 1 = Prevents BUC.TS (offset 3414, bit 0) and GCS.BBS (offset 3410h, bits 11:10) from being changed. This bit can only be written from 0 to 1 once.			RWLO

7.1.1.57 Offset 3414h: BUC—Backed Up Control Register (B0:D31:F0)

All bits in this register are in the RTC well and only cleared by RTCRST#.

Table 7-58. Offset 3414h: BUC—Backed Up Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3414h **Offset End:** 3414h	
Size: 8 bit	**Default:** 0000000xb			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	Reserved	Reserved			
05	Reserved	Reserved			RW
04	SDO	Daylight Savings Override: 0 = Daylight Savings is Enabled. 1 = The DSE bit in RTC Register B is set to Read-only with a value of 0 to disable daylight savings.			RW
03 :01	Reserved	Reserved			
00	TS	Top Swap: • When set, PCH will invert either A16, A17, or A18 for cycles going to the BIOS space (but not the feature space) in the FWH. • When cleared, PCH will not invert A16. If booting from LPC (FWH), then the Boot Block size is 64KB and A16 is inverted if Top Swap is enabled. If booting from SPI, the BOOT_BLOCK_SIZE soft strap determines if A16, A17, or A18 should be inverted if Top Swap is enabled. If PCH is strapped for Top-Swap (GNT[3]# is low at rising edge of PWROK), this bit cannot be cleared by software. The strap jumper should be removed and the system rebooted.			RW

7.1.1.58 Offset 3418h: FD—Function Disable Register (B0:D31:F0)

When disabling a function, only the configuration space is disabled. Software must ensure that all functionality within a controller that is not desired (such as memory spaces, I/O spaces, and DMA engines) is disabled prior to disabling the function.

When a function is disabled, software must not attempt to re-enable it. A disabled function can only be re-enabled by a platform reset.

Table 7-59. Offset 3418h: FD—Function Disable Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3418h **Offset End:** 341Bh	
Size: 32 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :26	Reserved	Reserved			
25	SAD2	Serial ATA Disable 2 — Default is 0: 0 = The SATA* controller #2 (D31:F5) is enabled. 1 = The SATA* controller #2 (D31:F5) is disabled.		0	RW
24	TTD	Thermal Throttle Disable — Default is 0: 0 = Thermal Throttle is enabled. 1 = Thermal Throttle is disabled.		0	RW
23 :20	Reserved	Reserved			
19	PE4D	PCI Express* 4 Disable — Default is 0. When disabled, the link for this port is put into the "link down" state: 0 = PCI Express* port #4 is enabled. 1 = PCI Express* port #4 is disabled. When disabled, the link for this port is put into the "link down" state.		0	RW
18	PE3D	PCI Express* 3 Disable — Default is 0. When disabled, the link for this port is put into the link down state: 0 = PCI Express* port #3 is enabled. 1 = PCI Express* port #3 is disabled. When disabled, the link for this port is put into the "link down" state.		0	RW
17	PE2D	PCI Express* 2 Disable — Default is 0. When disabled, the link for this port is put into the "link down" state: 0 = PCI Express* port #2 is enabled. 1 = PCI Express* port #2 is disabled. When disabled, the link for this port is put into the "link down" state.		0	RW
16	PE1D	PCI Express* 1 Disable — Default is 0. When disabled, the link for this port is put into the link down state: 0 = PCI Express* port #1 is enabled. 1 = PCI Express* port #1 is disabled. When disabled, the link for this port is put into the "link down" state.		0	RW
15	U2D	USB2 Disable: When set, the USB2 host controller is disabled.		0	RW

Table 7-59. Offset 3418h: FD—Function Disable Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** RCBA		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 3418h **Offset End:** 341Bh	
Size: 32 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
14	LPD	LPC Bridge Disable: When set, the LPC bridge is disabled. Unlike the other disables in this register, the following additional spaces will no longer be decoded by the LPC bridge: • Memory cycles below 16MB (1000000h) • I/O cycles below 64kB (10000h) Memory cycles in the LPC BIOS range below 4GB will still be decodedwhen this bit is set, but the aliases at the top of 1MB (the E and F segment) no longer will be decoded. The decode for the internal I/OxAPIC memory ranges and theEOI message are qualified by the APIC Enable (AEN) bit, but not the LPC Bridge Disable (LPD) bit.		0	RW
13 :04	Reserved	Reserved			
03	SD	SM Bus Disable: When set, the SM Bus controller is disabled. This only disables the configuration space.		0	RW
02	SAD1	Serial ATA Disable 1: When set, the serial SATA* controller 1 (SATA*1, Device 31, Function 2) is disabled.		0	RW
01 :00	Reserved	Reserved			

7.1.1.59 Offset 341Ch: CG—Clock Gating (B0:D31:F0)

Table 7-60. Offset 341Ch: CG—Clock Gating

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 341Ch **Offset End:** 341Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:31	LPC	Legacy Dynamic Clock Gate Enable: 0 = Legacy Dynamic Clock Gating is Disabled 1 = Legacy Dynamic Clock Gating is Enabled			RW
30 :20	Reserved	Reserved			
19		USB* EHCI Static Clock Gate Enable: 0 = USB* EHCI Static Clock Gating is Disabled 1 = USB* EHCI Static Clock Gating is Enabled			RW
18		USB* EHCI Dynamic Clock Gate Enable: 0 = USB* EHCI Dynamic Clock Gating is Disabled 1 = USB* EHCI Dynamic Clock Gating is Enabled			RW
17		SATA* Port 5 Dynamic Clock Gate Enable: 0 = SATA* Port 5 Dynamic Clock Gating is Disabled 1 = SATA* Port 5 Dynamic Clock Gating is Enabled			RW
16		SATA* Port 4 Dynamic Clock Gate Enable: 0 = SATA* Port 4 Dynamic Clock Gating is Disabled 1 = SATA* Port 4 Dynamic Clock Gating is Enabled			RW
15 :06	Reserved	Reserved			
05	SMBCGEN	SMBus Clock Gating EnablE: 0 = SMBus Clock Gating is Disabled. 1 = SMBus Clock Gating is Enabled.			RW
04 :01	Reserved	Reserved			
00		PCI Express* Root Port Static Clock Gate Enable: 0 = PCI Express* root port Static Clock Gating is Disabled 1 = PCI Express* root port Static Clock Gating is Enabled			RW

7.1.1.60 Offset 3420h: FDSW—Function Disable SUS Well (B0:D31:F0)

Table 7-61. Offset 3420h: FDSW—Function Disable SUS Well

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3420h **Offset End:** 3420h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	FDSWL	Function Disable SUS Well Lockdown: 0 = FDSW registers are not locked down 1 = FDSW registers are locked down			RW
06 :00	Reserved	Reserved			

7.1.1.61 Offset 3428h: FD2—Function Disable 2 (B0:D31:F0)

Table 7-62. Offset 3428h: FD2—Function Disable 2

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3428h **Offset End:** 342Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04	KTD	KT Disable — Default is 0: 0 = Keyboard Text controller (D22:F3) is enabled. 1 = Keyboard Text controller (D22:F3) is Disabled			RW
03	IRERD	IDE-R Disable — Default is 0: 0 = IDE Redirect controller (D22:F2) is Enabled. 1 = IDE Redirect controller (D22:F2) is Disabled.			RW
02	MEI2D	Intel(R) MEI #2 Disable — Default is 0: 0 = Intel(R) MEI controller #2 (D22:F1) is enabled. 1 = Intel(R) MEI controller #2 (D22:F1) is disabled.			RW
01	MEI1D	Intel(R) MEI #1 Disable — Default is 0: 0 = Intel(R) MEI controller #1 (D22:F0) is enabled. 1 = Intel(R) MEI controller #1 (D22:F0) is disabled.			RW
00	Reserved	Reserved			

7.1.1.62 Offset 3500h: USBIR[0:5]—USB Initialization Register [0-5]

Address Offset:
Port 0: 3500h
Port 1: 3504h
Port 2: 3508h
Port 3: 350Ch
Port 4: 3510h
Port 5: 3514h

These registers are located in the Suspend well.

Table 7-63. Offset 3500h: USBIR[0:5]—USB Initialization Register [0-5]

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3500h **Offset End:** 3514h	
Size: 32 bit	**Default:** 20000B4Ah			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11 :00		USB* Port[n] Initialization Field 1 — Configures the USB* transmitter of port n. This field must be programmed by BIOS. All other values are reserved. Note: See the platform design collateral for description of topology definitions. C4Bh Long Topology Setting (10 -14 inches) - This setting should be used for desktop back panel ports, long mobile back panel ports and long mobile docking ports where increased transmit amplitude is required to optimize USB* transmit characteristics. Note: This setting may also be used for short topology ports that require an increased transmit amplitude. This setting should not be used for short topology ports that have sufficient transmit amplitude when using the short topology setting. A4Bh Short Topology Setting (< 10 inches)- This setting should be used for most desktop front panel ports, short mobile back panel ports and short mobile docking ports for which decreased transmit amplitude is required to optimize USB* transmit characteristics. Note: This setting is weaker than the long topology setting and may not be strong enough for use with all short topology ports.			RW

7.1.1.63 Offset 3564h: USBIRC—USB Initialization Register C (B0:D31:F0)

Table 7-64. Offset 3564h: USBIRC—USB Initialization Register C

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3564h **Offset End:** 3567h		
Size: 32 bit	**Default:** 0000371Bh		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
08		USB* Initialization Register C Field 1. BIOS must program this field to 1b.			RW
07 :03		USB* Initialization Register C Field 2. BIOS must program this field to 10100b.			
02 :00	Reserved	Reserved			

7.1.1.64 Offset 3570h: USBIRA—USB Initialization Register A (B0:D31:F0)

Table 7-65. Offset 3570h: USBIRA—USB Initialization Register A

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 3570h **Offset End:** 3573h		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		USB* Initialization Register A. BIOS must program this field to 06200003h.			

7.1.1.65 Offset 357Ch: USBIRB—USB Initialization Register B (B0:D31:F0)

Table 7-66. Offset 357Ch: USBIRB—USB Initialization Register B

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 357Ch **Offset End:** 357Fh	
Size: 32 bit	**Default:** 0001C000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		USB* Initialization Register B. BIOS must program this field to 0001C000h.			

7.1.1.66 Offset 3590h: MISCCTL—Miscellaneous Control Register (B0:D31:F0)

This register is in the suspend well. This register is not reset on D3-to-D0, HCRESET nor core well reset.

Table 7-67. Offset 3590h: MISCCTL—Miscellaneous Control Register

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 3590h **Offset End:** 3594h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved			
01		EHCI 2 USBR Enable — When set, this bit enables support for the USB-r redirect device on the EHCI controller in Device 26. SW must complete programming the following registers before this bit is set: 1. UHCI function disables 2. HCCSPARAMS (N_CC, N_Ports)			RW
00		EHCI 1 USBR Enable — When set, this bit enables support for the USB-r redirect device on the EHCI controller in Device 29. SW must complete programming the following registers before this bit is set: 1. UHCI function disables 2. HCCSPARAMS (N_CC, N_Ports)			RW

7.1.1.67 Offset 359Ch: PDO—USB Port Disable Override

The register is implemented in the Suspend well to maintain the policy when the core power goes down. This register is not reset on D3-to-D0, HCRESET nor core well reset. Disabling Port 1 will block the Debug Port functionality for this EHCI.

Table 7-68. Offset 359Ch: PDO—USB Port Disable Override

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 359Ch **Offset End:** 359Fh	
Size: 32 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13 :00		USB* Port Disable — When RMH is disabled: 0 = Prevents the corresponding USB* port from reporting a Device Connection in the EHCI and UHCI Register Space. Attempts to enable the port through the PORTSC register will be ignored by the hardware when this bit is 1. 1 = Allows corresponding USB* port to report a Device Connection in the EHCI and UHCI Register Space. When RMH is enabled: 0 = Prevents the corresponding USB* port from reporting a Device Connection to the hub. Attempts to enable the port will be ignored by the hardware when this bit is 1. 1 = Allows corresponding USB* port to report a Device Connection the hub. This register can not be written when the USB* Per-Port Registers Write Enable bit (UPRWC register, PMBASE + 3C, bit 1) is 0.			RW

7.1.1.68 Offset 35A0h: USBOCM1—Overcurrent MAP Register 1 (B0:D31:F0)

Table 7-69. Offset 35A0h: USBOCM1—Overcurrent MAP Register 1

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 35A0h **Offset End:** 35A3h	
Size: 32 bit	**Default:** C0300C03h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24		OC3 Mapping — Each bit position maps OC3# to a set of ports as follows: The OC3# pin is ganged to the overcurrent signal of each port that has its corresponding bit set. It is SW's responsibility to ensure that a given port's bit map is set only for one OC pin. **Bit** 31 30 29 28 27 26 25 24 **Port** X X 5 4 3 2 1 0			
23 :16		OC2 Mapping — Each bit position maps OC2# to a set of ports as follows: The OC2# pin is ganged to the overcurrent signal of each port that has its corresponding bit set. It is SW's responsibility to ensure that a given port's bit map is set only for one OC pin. **Bit** 23 22 21 20 19 18 17 16 **Port** X X 5 4 3 2 1 0			
15 :08		OC1 Mapping — Each bit position maps OC1# to a set of ports as follows: The OC1# pin is ganged to the overcurrent signal of each port that has its corresponding bit set. It is SW's responsibility to ensure that a given port's bit map is set only for one OC pin. **Bit** 15 14 13 12 11 10 9 8 **Port** X X 5 4 3 2 1 0			
07 :00		OC0 Mapping — Each bit position maps OC0# to a set of ports as follows: The OC0# pin is ganged to the overcurrent signal of each port that has its corresponding bit set. It is SW's responsibility to ensure that a given port's bit map is set only for one OC pin. **Bit** 7 6 5 4 3 2 1 0 **Port** X X 5 4 3 2 1 0			

7.1.1.69 Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register (B0:D31:F0)

Table 7-70. Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 35B0h **Offset End:** 35B3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09		RMH 2 Inherit EHCI2 Wake Control Settings — When this bit is set, the RMH behaves as if bits 6:4 of this register reflect the appropriate bits of EHCI PORTSC0 bits 22:20.			
08		RMH 1 Inherit EHCI1 Wake Control Settings — When this bit is set, the RMH behaves as if bits 2:0 of this register reflect the appropriate bits of EHCI PORTSC0 bits 22:20.			
07		RMH 2 Upstream Wake on Device Resume — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables the port to be sensitive to device initiated resume events as system wake-up events. i.e, the hub will initiate a resume on its upstream port and cause a wake from Sx when a device resume occurs on an enabled DS port 1 = Device resume event is seen on a downstream port, the hub does not initiate a wake upstream and does not cause a wake from Sx			
06		RMH 2 Upstream Wake on OC Disable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables the port to be sensitive to over-current conditions as system wake-up events. i.e, the hub will initiate a resume on its upstream port and cause a wake from Sx when an OC condition occurs on an enabled DS port 1 = Over-current event does not initiate a wake upstream and does not cause a wake from Sx			
05		RMH 2 Upstream Wake on Disconnect Disable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables disconnect events on downstream port to be treated as resume events to be propagated upstream. In this case, it is allowed to initiate a wake on its upstream port and cause a system wake from Sx in response to a disconnect event on a downstream port 1 = Downstream disconnect events do not initiate a resume on its upstream port or cause a resume from Sx.			
04		RMH 2 Upstream Wake on Connect Enable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables connect events on a downstream port to be treated as resume events to be propagated upstream. As well as waking up the system from Sx. 1 = Downstream connect events do not wake the system from Sx nor does it initiate a resume on its upstream port.			

Table 7-70. Offset 35B0h: RMHWKCTL—Rate Matching Hub Wake Control Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** RCBA	**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 35B0h **Offset End:** 35B3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
03		RMH 1 Upstream Wake on Device Resume — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables the port to be sensitive to device initiated resume events as system wake-up events. i.e, the hub will initiate a resume on its upstream port and cause a wake from Sx when a device resume occurs on an enabled DS port 1 = Device resume event is seen on a downstream port, the hub does not initiate a wake upstream and does not cause a wake from Sx			
02		RMH 1 Upstream Wake on OC Disable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables the port to be sensitive to over-current conditions as system wake-up events. i.e, the hub will initiate a resume on its upstream port and cause a wake from Sx when an OC condition occurs on an enabled DS port 1 = Over-current event does not initiate a wake upstream and does not cause a wake from Sx			
01		RMH 1 Upstream Wake on Disconnect Disable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables disconnect events on downstream port to be treated as resume events to be propagated upstream. In this case, it is allowed to initiate a wake on its upstream port and cause a system wake from Sx in response to a disconnect event on a downstream port 1 = Downstream disconnect events do not initiate a resume on its upstream port or cause a resume from Sx.			
00		RMH 1 Upstream Wake on Connect Enable — This bit governs the hub behavior when globally suspended and the system is in Sx: 0 = Enables connect events on a downstream port to be treated as resume events to be propagated upstream. As well as waking up the system from Sx. 1 = Downstream connect events do not wake the system from Sx nor does it initiate a resume on its upstream port.			

§ §

8.0 SATA* Controller Registers (B0:D31:F2)

8.1 PCI Configuration Registers (SATA–B0:D31:F2)

Note: Address locations that are not shown should be treated as Reserved.

All of the SATA* registers are in the core well. None of the registers can be locked.

Note: SATA* Controller #1 (SATA - B0:D31:F2) supports two ports (Ports 4 and 5) in the AHCI mode.

8.1.1 SATA* Controller PCI Register Address Map

Table 8-1. SATA* Controller PCI Register Address Map (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: Vendor Identification Register (SATA–B0:D31:F2)" on page 422	8086h
02h	03h	"Offset 02h: Device Identification Register (SATA–B0:D31:F2)" on page 422	2323h/23A3h[1]
04h	05h	"Offset 04h: PCI Command Register (SATA–B0:D31:F2)" on page 423	0000h
06h	07h	"Offset 06h: PCI Status Register (SATA–B0:D31:F2)" on page 424	02B0h
08h	08h	"Offset 08h: Revision Identification Register (SATA–B0:D31:F2)" on page 425	See bit description
09h	09h	"Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 01h" on page 425	See bit description
0Ah	0Ah	"Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 06h" on page 426	01h
0Ah	0Ah	"Offset 0Ah: Sub Class Code Register (SATA–B0:D31:F2)" on page 427	See bit description
0Bh	0Bh	"Offset 0Bh: Base Class Code Register (B0:D31:F2)" on page 427	01h
0Dh	0Dh	"Offset 0Dh: Primary Master Latency Timer Register (B0:D31:F2)" on page 427	00h
0Eh	0Eh	"Offset 0Eh: Header Type (SATA–B0:D31:F2)" on page 428	80h
10h	13h	"Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F2)" on page 428	00000001h
14h	17h	"Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F2)" on page 429	00000001h
18h	1Bh	"Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1)" on page 429	00000001h
1Ch	1Fh	"Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F2)" on page 430	00000001h
20h	23h	"Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F2)" on page 430	00000001h
24h	27h	"Offset 24h: When SCC is Not 01h (SATA–B0:D31:F2)" on page 431	00000000h

Table 8-1. SATA* Controller PCI Register Address Map (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
24h	27h	"Offset 24h: When SCC is 01h (SATA–B0:D31:F2)" on page 432	00000001h
2Ch	2Dh	"Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F2)" on page 432	0000h
2Eh	2Fh	"Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F2)" on page 433	0000h
34h	34h	"Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F2)" on page 433	80h
3Ch	3Ch	"Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F2)" on page 433	00h
3Dh	3Dh	"Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F2)" on page 434	See register description
40h	43h	"Offset 40h: IDE Timing Register (SATA–B0:D31:F2)" on page 434	0000h
48h	48h	"Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F2)" on page 435	00h
4Ah	4Bh	"Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F2)" on page 435	0000h
54h	57h	"Offset 54h: IDE I/O Configuration Register (SATA–B0:D31:F2)" on page 436	00000000h
70h	71h	"Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F2)" on page 436	XX01h
72h	73h	"Offset 72h: PCI Power Management Capabilities Register (SATA–B0:D31:F2)" on page 437	X003h
74h	75h	"Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F2)" on page 438	XX08h
80h	81h	"Offset 80h: Message Signaled Interrupt Capability Identification (SATA–B0:D31:F2)" on page 439	7005h
82h	83h	"Offset 82h: Message Signaled Interrupt Message Control (SATA–B0:D31:F2)" on page 440	0000h
84h	87h	"Offset 84h: Message Signaled Interrupt Message Address (SATA–B0:D31:F2)" on page 441	00000000h
88h	89h	"Offset 88h: Message Signaled Interrupt Message Data (SATA–B0:D31:F2)" on page 441	0000h
90h	90h	"Offset 90h: MAP—Address Map Register (SATA–B0:D31:F2)" on page 442	0000h
92h	93h	"Offset 92h: PCS—Port Control and Status Register (SATA–B0:D31:F2)" on page 444	0000h
94h	97h	"Offset 94h: SCLKCG—SATA Clock Gating Control Register (SATA–B0:D31:F2)" on page 445	00000000h
9Ch	9Fh	"Offset 9Ch: SCLKGC—SATA Clock General Configuration Register (SATA–B0:D31:F2)" on page 446	00000000h
A0h	A0h	"Offset A0h: SIRI—SATA Indexed Registers Index (SATA–B0:D31:F2)" on page 447	00h
A4h	A7h	"Offset A4h: STRD—SATA Indexed Register Data (SATA–B0:D31:F2)" on page 447	XXXXXXXXh
A8h	ABh	"Offset A8h: SATACR0—SATA Capability Register 0 (SATA–B0:D31:F2)" on page 448	0010B012h
ACh	AFh	"Offset ACh: SATACR1—SATA Capability Register 1 (SATA–B0:D31:F2)" on page 449	00000048h
B0h	B1h	"Offset B0h: FLRCID—FLR Capability ID (SATA–B0:D31:F2)" on page 450	0009h
B2h	B3h	"Offset B2h: FRLCLV—FLR Capability Length and Version (SATA–B0:D31:F2)" on page 450	xx06h
B2h	B3h	"Offset B2h: FLRCLV—FLR Capability Length and Version (SATA–B0:D31:F2)" on page 451	xx06h
B4h	B5h	"Offset B4h: FLRC—FLR Control (SATA–B0:D31:F2)" on page 451	0000h
C0h	C0h	"Offset C0h: ATC—APM Trapping Control Register (SATA–B0:D31:F2)" on page 452	00h
C4h	C4h	"Offset C4h: ATS—APM Trapping Status Register (SATA–B0:D31:F2)" on page 452	00h
D0h	D0h	"Offset D0h: SP—Scratch Pad Register (SATA–B0:D31:F2)" on page 453	00000000h

Table 8-1. SATA* Controller PCI Register Address Map (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
E0h	E3h	"Offset E0h: BFCS—BIST FIS Control/Status Register (SATA–B0:D31:F2)" on page 453	00000000h
E4h	E7h	"Offset E4h: BFTD1—BIST FIS Transmit Data1 Register (SATA–B0:D31:F2)" on page 455	00000000h
E8h	EBh	"Offset E8h: BFTD2—BIST FIS Transmit Data2 Register (SATA–B0:D31:F2)" on page 455	00000000h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

Note: The PCH SATA* controller is not arbitrated as a PCI device, therefore it does not need a master latency timer.

8.1.1.1 Offset 00h: Vendor Identification Register (SATA–B0:D31:F2)

Table 8-2. Offset 00h: Vendor Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

8.1.1.2 Offset 02h: Device Identification Register (SATA–B0:D31:F2)

Table 8-3. Offset 02h: Device Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2323h/23A3h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device Identification		2323h/23A3h[1]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

8.1.1.3 Offset 04h: PCI Command Register (SATA–B0:D31:F2)

Table 8-4. Offset 04h: PCI Command Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 04h **Offset End:** 05h	
Size:	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		00h	
10	INTx_Disable	Interrupt Disable — This disables pin-based INTx# interrupts. This bit has no effect on MSI operation. 0 = Internal INTx# messages are generated if there is an interrupt and MSI is not enabled. 1 = Internal INTx# messages will not be generated.		0h	RW
09	FBE	Fast Back to Back Enable — Reserved as 0.		0h	RO
08	SERR_EN	SERR# Enable — Reserved as 0.		0h	RO
07	WCC	Wait Cycle Control — Reserved as 0.		0h	RO
06	PER	Parity Error Response: 0 = Disabled. SATA* controller will not generate PERR# when a data parity error is detected. 1 = Enabled. SATA* controller will generate PERR# when a data parity error is detected.		0h	RW
05	VPS	VGA Palette Snoop —Reserved as 0.		0h	RO
04	PMWE	Postable Memory Write Enable — Reserved as 0.		0h	RO
03	SCE	Special Cycle Enable —Reserved as 0.		0h	RO
02	BME	Bus Master Enable — This bit controls the PCH's ability to act as a PCI master for IDE Bus Master transfers. This bit does not impact the generation of completions for split transaction commands.		1	RW
01	MSE	Memory Space Enable — Controls access to the SATA* controller's target memory space (for AHCI). This bit is RO '0' when not in AHCI modes.		1	RW/RO
00	IOSE	I/O Space Enable — This bit controls access to the I/O space registers. 0 = Disables access to the Legacy or Native IDE ports (both Primary and Secondary) as well as the Bus Master I/O registers. 1 = Enable. The Base Address register for the Bus Master registers should be programmed before this bit is set.		1	RW

8.1.1.4 Offset 06h: PCI Status Register (SATA–B0:D31:F2)

For the writable bits, software must write a 1 to clear bits that are set. Writing a 0 to the bit has no effect.

Table 8-5. Offset 06h: PCI Status Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 02B0h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the Parity Error Response bit (B0:D31:D2; Offset 04h bit 6) is not set.			RWC
14	SSE	Signaled System Error — Set when the LPC bridge signals a system error to the internal SERR# logic.			RWC
13	RMA	Received Master Abort: 0 = No master abort received. 1 = Set when the bridge receives a master abort status from the I/O data bus.			RWC
12	RTA	Reserved as 0			RO
11	STA	Signaled Target Abort.			RO
10 :09	DEV_STS	DEVSEL# Timing Status: 01 = Hardwired; Controls the device select time for the SATA* controller's PCI interface.			RO
08	DPED	Data Parity Error Detected — For PCH, this bit can only be set on read completions received from the bus when there is a parity error. SATA* controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set.			RWC
07 :05	FB2BC	Fast Back to Back Capable — Reserved as 1.			RO
04	UDF	User Definable Features — Reserved as 0.			RO
03	66MHZ_CAP	66MHz Capable — Reserved as 1.			RO
02 :00	CAP_LIST	Capabilities List — This bit indicates the presence of a capabilities list. The minimum requirement for the capabilities list must be PCI power management for the SATA* controller.			RO

8.1.1.5 Offset 08h: Revision Identification Register (SATA–B0:D31:F2)

Table 8-6. Offset 08h: Revision Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See bit description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: 8-bit value indicating the stepping of the SATA* Controller hardware		Variable	RO

8.1.2 Programming Interface Register (SATA–B0:D31:F2)

8.1.2.1 Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 01h

Table 8-7. Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 01h (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** See bit description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		This read-only bit is a 1 to indicate that the PCH supports bus master operation			RO
06 :04	Reserved	Reserved. Will always return 0.			
03	SNC	Secondary Mode Native Capable: 0 = Secondary controller only supports legacy mode. 1 = Secondary controller supports both legacy and native modes. When MAP.MV (B0:D31:F2:Offset 90:bits 1:0) is any value other than 00b, this bit reports as a 0. When MAP.MV is 00b, this bit reports as a 1.			RO

Table 8-7. Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 01h (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31 :F2	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** See bit description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	SNE	Secondary Mode Native Enable: Determines the mode that the secondary channel is operating in. 0 = Secondary controller operating in legacy (compatibility) mode 1 = Secondary controller operating in native PCI mode. When MAP.MV (B0:D31:F2:Offset 90:bits 1:0) is any value other than 00b, this bit is read-only (RO). When MAP.MV is 00b, this bit is read/write (R/W). If this bit is set by software, then the PNE bit (bit 0 of this register) must also be set by software. While in theory these bits can be programmed separately, such a configuration is not supported by hardware.			RW
01	PNC	Primary Mode Native Capable: 0 = Primary controller only supports legacy mode. 1 = Primary controller supports both legacy and native modes. When MAP.MV (B0:D31:F2:Offset 90:bits 1:0) is any value other than 00b, this bit reports as a 0. When MAP.MV is 00b, this bit reports as a 1			RO
00	PNE	Primary Mode Native Enable: Determines the mode that the primary channel is operating in. 0 = Primary controller operating in legacy (compatibility) mode. 1 = Primary controller operating in native PCI mode. If this bit is set by software, then the SNE bit (bit 2 of this register) must also be set by software simultaneously.			RW

8.1.2.2 Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 06h

Table 8-8. Offset 0Ah: When Sub Class Code Register (B0:D31:F2:Offset 0Ah) = 06h

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31 :F2	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IF	Interface - Indicates the SATA* Controller supports AHCI, Revision 1.2.			RO

8.1.2.3 Offset 0Ah: Sub Class Code Register (SATA–B0:D31:F2)

Table 8-9. Offset 0Ah: Sub Class Code Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** See bit description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code: This field specifies the sub-class code of the controller, per the table below:			RO

8.1.2.4 Offset 0Bh: Base Class Code Register (B0:D31:F2)

Table 8-10. Offset 0Bh: Base Class Code Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code: 01h = Mass storage device			RO

8.1.2.5 Offset 0Dh: Primary Master Latency Timer Register (B0:D31:F2)

Table 8-11. Offset 0Dh: Primary Master Latency Timer Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MLTC	Master Latency Timer Count: 00h = Hardwired. The SATA* controller is implemented internally, and is not arbitrated as a PCI device, so it does not need a Master Latency Timer.			RO

8.1.2.6 Offset 0Eh: Header Type (SATA–B0:D31:F2)

Table 8-12. Offset 0Eh: Header Type (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 80h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	MFD	Multi-function Device: Indicates this SATA* controller is not part of a multifunction device.			RO
06 :00	HL	Header Layout: Indicates that the SATA* controller uses a target device layout.			RO

8.1.2.7 Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F2)

Note: This 8-byte I/O space is used in native mode for the Primary Controller's Command Block.

Table 8-13. Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :03	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
02 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

8.1.2.8 Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F2)

Note: This 4-byte I/O space is used in native mode for the Primary Controller's Command Block.

Table 8-14. Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :02	BA	Base Address — This field provides the base address of the I/O space (4 consecutive I/O locations).			RW
01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

8.1.2.9 Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1)

Note: This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

Table 8-15. Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :03	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
02 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

8.1.2.10 Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F2)

Note: This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

Table 8-16. Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :02	BA	Base Address — This field provides the base address of the I/O space (4 consecutive I/O locations).			RW
01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

8.1.2.11 Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F2)

The Bus Master IDE interface function uses Base Address register 5 to request a 16-byte IO space to provide a software interface to the Bus Master functions. Only 12 bytes are actually used (6 bytes for primary, 6 bytes for secondary). Only bits [15:4] are used to decode the address.

Table 8-17. Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :05	BA	Base Address — This field provides the base address of the I/O space (16 consecutive I/O locations).			RW
04		Base — When SCC is 01h, this bit will be R/W resulting in requesting 16B of I/O space. When SCC is not 01h, this bit will be Read Only 0, resulting in requesting 32B of I/O space.			RW/RO
03 :01	Reserved	Reserved			RO
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

8.1.3 AHCI Base Address Register/Serial ATA Index Data Pair Base Address (SATA–B0:D31:F2)

When the programming interface is not IDE (for example, SCC is not 01h), this register is named ABAR. When the programming interface is IDE, this register becomes SIDPBA.

Hardware does not clear those BA bits when switching from IDE component to non-IDE component or vice versa. BIOS is responsible for clearing those bits to 0 since the number of writable bits changes after component switching (as indicated by a change in SCC). In the case, this register will then have to be re-programmed to a proper value.

8.1.3.1 Offset 24h: When SCC is Not 01h (SATA–B0:D31:F2)

When the programming interface is not IDE, the register represents a memory BAR allocating space for the AHCI memory registers defined in Section 8.4.

Note: The ABAR register must be set to a value of 0001_0000h or greater.

Table 8-18. Offset 24h: When SCC is Not 01h (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :11	BA	Base Address — Base address of register memory space (aligned to 1 KB)			RW
10 :04	Reserved	Reserved			
03	PF	Prefetchable — Indicates that this range is not pre-fetchable			RO
02 :01	TP	Type — Indicates that this range can be mapped anywhere in 32-bit address space.			RO
00	RTE	Resource Type Indicator — Hardwired to 0 to indicate a request for register memory space.			RO

8.1.3.2 Offset 24h: When SCC is 01h

When the programming interface is IDE, the register becomes an I/O BAR allocating 16 bytes of I/O space for the I/O-mapped registers. Although 16 bytes of locations are allocated, only 8 bytes are used to as SINDX and SDATA registers; with the remaining 8 bytes preserved for future enhancement.

Table 8-19. Offset 24h: When SCC is 01h (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :04	BA	Base Address — Base address of the I/O space.			RW
03 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Indicates a request for I/O space.			RO

8.1.3.3 Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F2)

Table 8-20. Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	Subsystem Vendor ID — Value is written by BIOS. No hardware action taken on this value.			RWO

8.1.3.4 Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F2)

Table 8-21. Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	Subsystem Vendor ID — Value is written by BIOS. No hardware action taken on this value.			RWO

8.1.3.5 Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F2)

Table 8-22. Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 80h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CAP_PTR	Capabilities Pointer — Indicates that the first capability pointer offset is 80h. This value changes to 70h if the Sub Class Code (SCC) (Dev 31:F2:0Ah) is configure as IDE mode (value of 01).			RO

8.1.3.6 Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F2)

Table 8-23. Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IL	Interrupt Line — This field is used to communicate to software the interrupt line that the interrupt pin is connected to. Interrupt Line register is not reset by FLR			RW

8.1.3.7 Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F2)

Table 8-24. Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IP	Interrupt Pin — This reflects the value of D31IP.SIP (Chipset Config Registers:Offset 3100h:bits 11:8).			RO

8.1.3.8 Offset 40h: IDE Timing Register (SATA–B0:D31:F2)

Address Offset: Primary: 40h–41h

Secondary Offset: 42h–43h

Table 8-25. Offset 40h: IDE Timing Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 40h **Offset End:** 43h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	IDE	IDE Decode Enable — Individually enable/disable the Primary or Secondary decode. 0 = Disable. 1 = Enables the PCH to decode the associated Command Blocks (1F0–1F7h for primary, 170–177h for secondary) and Control Block (3F6h for primary and 376h for secondary). This bit effects the IDE decode ranges for both legacy and native-Mode decoding. Note: This bit affects SATA operation in both combined and non-combined ATA modes.			RW
14 :00	Reserved	Reserved			

8.1.3.9 Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F2)

Address Offset: Primary: 48h

Table 8-26. Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 48h **Offset End:** 48h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	IReserved			
03 :00	SDMA_CNT	**SDMA_CNT Field 1**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

8.1.3.10 Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F2)

Address Offset: Primary: 4Ah–4Bh

Table 8-27. Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 4Ah **Offset End:** 4Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13 :12	SDMA_TIM4	**SDMA_TIM Field 4**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
11 :10	Reserved	Reserved			
09 :08	SDMA_TIM3	**SDMA_TIM Field 3**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
07 :06	Reserved	Reserved			
05 :04	SDMA_TIM2	**SDMA_TIM Field 2**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
03 :02	Reserved	Reserved			
01 :00	SDMA_TIM1	**SDMA_TIM Field 1**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

8.1.3.11 Offset 54h: IDE I/O Configuration Register (SATA–B0:D31:F2)

Address Offset: Primary: 54h–57h

Table 8-28. Offset 4Ah: IDE I/O Configuration Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :12	IDE_CFG2	**IDE_CONFIG Field 2**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
11 :08	Reserved	Reserved			
07 :00	IDE_CFG1	**IDE_CONFIG Field 1**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

8.1.4 PCI Power Management Capabilities

8.1.4.1 Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F2)

Table 8-29. Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 70h **Offset End:** 71h	
Size: 16 bit	**Default:** XX01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability: B0h — if SCC = 01h (IDE mode) indicating next item is FLR capability pointer. A8h — for all other values of SCC to point to the next capability structure.			RO
07 :00	CID	Capability ID — Indicates that this pointer is a PCI power management.			RO

8.1.4.2 Offset 72h: PCI Power Management Capabilities Register (SATA–B0:D31:F2)

Table 8-30. Offset 72h: PCI Power Management Capabilities Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 72h **Offset End:** 73h	
Size: 16 bit	**Default:** X003h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	PME_SUP	PME Support: 00000 = If SCC = 01h, indicates no PME support in IDE mode. 01000 = If SCC is not 01h, in a non-IDE mode, indicates PME# can be generated from the $D3_{HOT}$ state in the SATA* host controller.			RO
10	D2_SUP	D2 Support — Hardwired to 0. The D2 state is not supported			RO
09	D1_SUP	D1 Support — Hardwired to 0. The D1 state is not supported			RO
08 :06	AUX_CUR	Auxiliary Current — PME# from $D3_{COLD}$ state is not supported, therefore this field is 000b.			RO
05	DSI	Device Specific Initialization — Hardwired to 0 to indicate that no device-specific initialization is required.			RO
04	Reserved	Reserved			
03	PME_CLK	PME Clock — Hardwired to 0 to indicate that PCI clock is not required to generate PME#.			RO
02 :00	VER	Version — Hardwired to 011 to indicates support for Revision 1.2 of the PCI Power Management Specification.			RO

8.1.4.3 Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F2)

Table 8-31. Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 74h **Offset End:** 75h	
Size: 16 bit	**Default:** XX08h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	PMES	PME Status — Bit is set when a PME event is to be requested, and if this bit and PMEE is set, a PME# will be generated from the SATA* controller Note: Whenever SCC = 01h, hardware will automatically change the attribute of this bit to RO '0'. Software is advised to clear PMEE and PMES together prior to changing SCC thru MAP.SMS. This bit is not reset by Function Level Reset.			RWC
14 :09	Reserved	Reserved			
08	PMEE	PME Enable — When set, the SATA* controller generates WAKE# from $D3_{HOT}$ on a wake event. Note: Whenever SCCSCC = 01h, hardware will automatically change the attribute of this bit to RO '0'. Software is advised to clear PMEE and PMES together prior to changing SCC thru MAP.SMS. This bit is not reset by Function Level Reset.			RW
07 :04	Reserved	Reserved			
03	NSFRST	No Soft Reset — These bits are used to indicate whether devices transitioning from $D3_{HOT}$ state to D0 state will perform an internal reset. 0 = Device transitioning from $D3_{HOT}$ state to D0 state perform an internal reset. 1 = Device transitioning from $D3_{HOT}$ state to D0 state do not perform an internal reset. Configuration content is preserved. Upon transition from the $D3_{HOT}$ state to D0 state initialized state, no additional operating system intervention is required to preserve configuration context beyond writing to the PowerState bits. Regardless of this bit, the controller transition from $D3_{HOT}$ state to D0 state by a system or bus segment reset will return to the state D0 uninitialized with only PME context preserved if PME is supported and enabled.			RO
02	Reserved	Reserved			
01 :00	PS	Power State — These bits are used both to determine the current power state of the SATA* controller and to set a new power state. 00 = D0 state 11 = $D3_{HOT}$ state When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked.			RW

8.1.5 Message Signaled Interrupt Capability

The following set of MSI registers are only applicable to AHCI SKUs. There is no support for MSI when the software is operating in legacy (IDE) mode when AHCI is not enabled. Prior to switching from AHCI to IDE mode, software **must** make sure that MSI is disabled.

8.1.5.1 Offset 80h: Message Signaled Interrupt Capability Identification (SATA–B0:D31:F2)

There is no support for MSI when the software is operating in legacy (IDE) mode when AHCI is not enabled. Prior to switching from AHCI to IDE mode, software **must** make sure that MSI is disabled.

Table 8-32. Offset 80h: Message Signaled Interrupt Capability Identification (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 80h **Offset End:** 81h		
Size: 16 bit	**Default:** 7005h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Pointer — Indicates the next item in the list is the PCI power management pointer.			RO
07 :00	CID	Capability ID — Capabilities ID indicates MSI.			RO

8.1.5.2 Offset 82h: Message Signaled Interrupt Message Control (SATA–B0:D31:F2)

There is no support for MSI when the software is operating in legacy (IDE) mode when AHCI is not enabled. Prior to switching from AHCI to IDE mode, software **must** make sure that MSI is disabled.

Table 8-33. Offset 82h: Message Signaled Interrupt Message Control (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	Reserved	Reserved			
07	C64	64 Bit Address Capable: Capable of generating a 32-bit message only.			RO
06 :04	MME	Multiple Message Enable: When this field is cleared to '000' (and MSIE is set), only a single MSI message will be generated for all SATA* ports, and bits[15:0] of the message vector will be driven from MD[15:0].			RO
03 :01	MMC	Multiple Message Capable — Indicates the number of interrupt messages supported by the PCH SATA* controller. 000 =1 MSI Capable (When SCC bit is set to 01h. MSI is not supported in IDE mode) 100 = 8 MSI Capable			RO
00	MSIE	MSI Enable — If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. This bit is RW when SC.SCC is not 01h and is read-only 0 when SCC is 01h. CMD.ID bit has no effect on MSI. ***Note:*** Software must clear this bit to '0' to disable MSI first before changing the number of messages allocated in the MMC field. Software must also make sure this bit is cleared to '0' when operating in legacy mode (when GHC.AE = 0).			RW/RO

8.1.5.3 Offset 84h: Message Signaled Interrupt Message Address (SATA–B0:D31:F2)

There is no support for MSI when the software is operating in native (IDE) mode when AHCI is not enabled. Prior to switching from AHCI to IDE mode, software must make sure that MSI is disabled.

Table 8-34. Offset 84h: Message Signaled Interrupt Message Address (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address — Lower 32 bits of the system specified message address, always DWORD aligned.			RW
01 :00	Reserved	Reserved			

8.1.5.4 Offset 88h: Message Signaled Interrupt Message Data (SATA–B0:D31:F2)

There is no support for MSI when the software is operating in legacy (IDE) mode when AHCI is not enabled. Prior to switching from AHCI to IDE mode, software must make sure that MSI is disabled.

Table 8-35. Offset 88h: Message Signaled Interrupt Message Data (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 88h **Offset End:** 89h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	Data — this 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word of the data bus of the MSI memory write transaction. When the MME field is set to '001' or '010', bit [0] and bits [1:0] respectively of the MSI memory write transaction will be driven based on the source of the interrupt rather than from MD[2:0]. See the description of the MME field.			RW

8.1.5.5 Offset 90h: MAP—Address Map Register (SATA–B0:D31:F2)

Table 8-36. Offset 90h: MAP—Address Map Register (SATA–B0:D31:F2) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 90h **Offset End:** 90h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			
14 :08	SPD	**SATA* Port Disable (SPD)**. This register provides a bit corresponding to each of the SATA* port on the SATA*1 controller. A '1' in a bit position corresponding to the SATA* port prevents the SATA* port from being enabled via config PCS.PxE. Write of '1' to PCS.PxE has no effect when the corresponding SPD[x] bit is '1' Bit assignment to port: SPD[0] corresponds to SATA* port 0; SPD[1] to port 1 and so on. In preventing a port(s) from being enabled, BIOS shall first configures MAP.SPD. And only then BIOS configures the PCS.PxE. In order to securely prevent port 4 and/or 5 from enabled, BIOS shall ensure: • port 4 - D31F2 MAP[12]='1' and D31F5 MAP[8]='1' • port 5 - D31F2 MAP[13]='1' and D31F5 MAP[9]='1' This field is not reset by FLR.			RWO
07 :06	SMS	SATA* Mode Select — SW programs these bits to control the mode in which the SATA* Controller should operate: 00b = IDE mode 01b = AHCI mode 10b = Reserved 11b = Reserved ***Note:*** 1. The SATA* Function Device ID will change based on the value of this register. 2. When switching from AHCI mode to IDE mode, a 2 port SATA* controller (Device 31, Function 5) will be enabled. 3. AHCI mode may only be selected when MV = 00 4. 4. Programming these bits with values that are invalid will result in indeterministic behavior by the HW 5. 5. SW shall not manipulate SMS during runtime operation; for example the OS will not do this. The BIOS may choose to switch from one mode to another during POST. These bits are not reset by Function Level Reset.			RW

Table 8-36. Offset 90h: MAP—Address Map Register (SATA–B0:D31:F2) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 90h **Offset End:** 90h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05	SC	SATA* Port-to-Controller Configuration — This bit changes the number of SATA* ports available within each SATA* Controller. 0 = Up to 4 SATA* ports are available for Controller 1 (Device 31 Function 2) with ports [3:0] and up to 2 SATA* ports are available for Controller 2 (Device 31 Function 5) with ports [5:4]. 0 = Up to 6 SATA* ports are available for Controller 1 (Device 31 Function 2) with ports [5:0] and no SATA* ports are available for Controller 2 (Device 31 Function 5). This bit should be set to 1 only in AHCI mode. This bit is not reset by Function Level Reset.			RW
04 :02	Reserved	Reserved			
01 :00	MV	Map Value — Reserved			

8.1.5.6 Offset 92h: PCS - Port Control and Status Register (SATA–B0:D31:F2)

By default, the SATA* ports are set to the disabled state (bits [5:0] = '0'). When enabled by software, the ports can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the "off" state and cannot detect any devices.

If an AHCI-aware enabled operating system is being booted, then system BIOS shall insure that all supported SATA* ports are enabled prior to passing control to the OS. Once the AHCI aware OS is booted it becomes the enabling/disabling policy owner for the individual SATA* ports. This is accomplished by manipulating a port's PxSCTL and PxCMD fields. Because an AHCI aware OS will typically not have knowledge of the PxE bits and because the PxE bits act as master on/off switches for the ports, pre-boot software must insure that these bits are set to '1' prior to booting the OS, regardless of whether or not a device is currently on the port.

Table 8-37. Offset 92h: PCS—Port Control and Status Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 92h **Offset End:** 93h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	ORM	OOB Retry Mode: 0 = The SATA* controller will not retry after an OOB failure 1 = The SATA* controller will continue to retry after an OOB failure until successful (infinite retry)			RW
14	Reserved	Reserved			
13	P5P	Port 5 Present — The status of this bit may change at any time. This bit is cleared when the port is disabled via P5E. This bit is not cleared upon surprise removal of a device. 0 = No device detected. 1 = The presence of a device on Port 5 has been detected.			RO
12	P4P	Port 4 Present — The status of this bit may change at any time. This bit is cleared when the port is disabled via P4E. This bit is not cleared upon surprise removal of a device. 0 = No device detected. 1 = The presence of a device on Port 4 has been detected.			RO
11 :06	Reserved	Reserved			
05	P5E	Port 5 Enabled: 0 = Disabled. The port is in the 'off' state and cannot detect any devices. 1 = Enabled. The port can transition between the on, partial, and slumber states and can detect devices. Note: This bit takes precedence over P5CMD.SUD (offset ABAR+298h:bit 1) If MAP.SC is '0', if SCC is '01h' this bit will be read only '0' or if MAP.SPD[5] is '1'.			RW
04	P4E	Port 4 Enabled: 0 = Disabled. The port is in the 'off' state and cannot detect any devices. 1 = Enabled. The port can transition between the on, partial, and slumber states and can detect devices. Note: This bit takes precedence over P4CMD.SUD (offset ABAR+298h:bit 1) If MAP.SC is '0', if SCC is '01h' this bit will be read only '0' or if MAP.SPD[4] is '1'.			RW
03 :00	Reserved	Reserved			

8.1.5.7 Offset 94h: SCLKCG—SATA Clock Gating Control Register (SATA–B0:D31:F2)

Table 8-38. Offset 94h: SCLKCG—SATA Clock Gating Control Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30	Reserved	Reserved			
29 :28	PCD	Port Clock Disable: 0 = All clocks to the associated port logic will operate normally. 1 = The backbone clock driven to the associated port logic is gated and will not toggle. Bit 29: Port 5 Bit 28: Port 4 If a port is not available, software shall set the corresponding bit to 1. Software can also set the corresponding bits to 1 on ports that are disabled. Software cannot set the PCD [port x]='1' if the corresponding PCS.PxE='1' in either Dev31Func2 or Dev31Func5 (dual controller IDE mode).			RW
27 :09	Reserved	Reserved			
08 :00		SCLKCG Field 1 — BIOS must program these bits to 183h.			RW

8.1.5.8 Offset 9Ch: SCLKGC—SATA Clock General Configuration Register (SATA–B0:D31:F2)

Table 8-39. Offset 9Ch: SCLKGC—SATA Clock General Configuration Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 9Ch **Offset End:** 9Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	Reserved	Reserved			
06 :02		SATA* Traffic Monitor: 00000b = Disable. 00011b = Enable. SATA* Traffic Monitor allows for aggressive C2 Pop down by monitoring SATA* bus mastering activity. When enabled, BIOS must ensure bit 3 and bit 4 of Cx_STATE_CNF (Cx State Configuration Register) are ones. Note: This field is reset by PLTRST# and BIOS is required to reprogram the value after resuming from S3-S5. All other bit combinations are Reserved.			RW
01	SATA2PIND	SATA2-port Configuration Indicator: 0 = Normal configuration. 1 = One IDE Controller is implemented supporting only two ports for a Primary Master and a Secondary Master. Note: When set, BIOS must ensure that bit 2 and bit 3 of the AHCI PI registers are zeros. BIOS must also make sure that Port 2 and Port 3 are disabled (via PCS configuration register) and the port clocks are gated (via SCLKCG configuration register).			RO
00	SATA4PMIND	SATA4-port All Master Configuration Indicator: 0 = Normal configuration. 1 = Two IDE Controllers are implemented, each supporting two ports for a Primary Master and a Secondary Master. Note: When set, BIOS must ensure that bit 2 and bit 3 of the AHCI PI registers are zeros. BIOS must also make sure that Port 2 and Port 3 are disabled (via PCS configuration register) and the port clocks are gated (via SCLKCG configuration register).			RO

8.1.5.9 Offset A0h: SIRI—SATA Indexed Registers Index (SATA–B0:D31:F2)

This register provides access to the registers in SATA* Indexed Registers.

Table 8-40. Offset A0h: SIRI—SATA Indexed Registers Index (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** A0h **Offset End:** A0h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :02	IDX	Index — This field is a 5-bit index pointer into the SATA* Indexed Register space. Data is written into and read from the SIRD register (B0:D31:F2:A4h).			RW
01 :00	Reserved	Reserved			

8.1.5.10 Offset A4h: STRD—SATA Indexed Register Data (SATA–B0:D31:F2)

This register is used to read/write the register written in the Offset A0h: SATA* Indexed Registers Index (SATA-B0:D31:F2).

Table 8-41. Offset A4h: STRD—SATA Indexed Register Data (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** XXXXXXXXh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DTA	Data — 32-bit data value that is written to the register pointed to by SIRI (B0:D31:F2:A0h) or read from the register pointed to by SIRI.			RW

8.1.5.11 Offset A8h: SATACR0—SATA Capability Register 0 (SATA–B0:D31:F2)

This register shall be read-only 0 when SCC is 01h.

Table 8-42. Offset A8h: SATACR0—SATA Capability Register 0 (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** A8h **Offset End:** ABh	
Size: 32 bit	**Default:** 0010B012h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :20	MAJREV	Major Revision — Major revision number of the SATA* Capability Pointer implemented.			RO
19 :16	MINREV	Minor Revision — Minor revision number of the SATA* Capability Pointer implemented.			RO
15 :08	NEXT	Next Capability Pointer — Points to the next capability structure. These bits are not reset by Function Level Reset.			RWO
07 :00	CAP	Capability ID — This value of 12h has been assigned by the PCI SIG to designate the SATA* Capability Structure.			RO

8.1.5.12 Offset ACh: SATACR1—SATA Capability Register 1 (SATA–B0:D31:F2)

This register shall be read-only 0 when SCC is 01h.

Table 8-43. Offset ACh: SATACR1—SATA Capability Register 1 (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** ACh **Offset End:** AFh	
Size: 32 bit	**Default:** 00000048h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :04	BAROFST	BAR Offset — Indicates the offset into the BAR where the Index/Data pair are located (in Dword granularity). The Index and Data I/O registers are located at offset 10h within the I/O space defined by LBAR. A value of 004h indicates offset 10h. 000h = 0h offset 001h = 4h offset 002h = 8h offset 003h = Bh offset 004h = 10h offset ... FFFh = 3FFFh offset (max 16KB)			RO
03 :00	BARLOC	BAR Location — Indicates the absolute PCI Configuration Register address of the BAR containing the Index/Data pair (in Dword granularity). The Index and Data I/O registers reside within the space defined by LBAR in the SATA* controller. A value of 8h indicates offset 20h, which is LBAR. 0000 – 0011b = reserved 0100b = 10h => BAR0 0101b = 14h => BAR1 0110b = 18h => BAR2 0111b = 1Ch => BAR3 1000b = 20h => LBAR 1001b = 24h => BAR5 1010 – 1110b = reserved 1111b = Index/Data pair in PCI Configuration space. This isn't supported in the PCH.			RO

8.1.5.13 Offset B0h: FLRCID—FLR Capability ID (SATA–B0:D31:F2)

Table 8-44. Offset B0h: FLRCID—FLR Capability ID (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** B0h **Offset End:** B1h	
Size: 16 bit	**Default:** 0009h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08		Next Capability Pointer — 00h indicates the final item in the capability list.			RO
07 :00		Capability ID — The value of this field depends on the FLRCSSEL bit. 13h = If PFLRCSSEL = 0 09h (Vendor Specific) = If PFLRCSSEL = 1			RO

8.1.5.14 Offset B2h: FLRCLV—FLR Capability Length and Version (SATA–B0:D31:F2)

When FLRCSSEL = '0', this register is defined as follows:

Table 8-45. Offset B2h: FRLCLV—FLR Capability Length and Version (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** xx06h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09		FLR Capability — Support for Function Level reset. This bit is not reset by the Function Level Reset.			RWO
08		TXP Capability — Support for Transactions Pending (TXP) bit. TXP must be supported if FLR is supported.			RWO
07 :00		Vendor-Specific Capability ID — This field indicates the # of bytes of this Vendor Specific capability as required by the PCI specification. It has the value of 06h for the FLR capability.			RO

When FLRCSSEL = ‘1’, this register is defined as follows:

Table 8-46. Offset B2h: FLRCLV—FLR Capability Length and Version (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** xx06h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12		Vendor-Specific Capability ID — A value of 2h identifies this capability as the Function Level Reset (FLR).			RO
11 :08		Capability Version — This field indicates the version of the FLR capability.			RO
07 :00		Vendor-Specific Capability ID — This field indicates the # of bytes of this Vendor Specific capability as required by the PCI specification. It has the value of 06h for the FLR capability.			RO

8.1.5.15 Offset B4h: FLRC—FLR Control (SATA–B0:D31:F2)

Table 8-47. Offset B4h: FLRC—FLR Control (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** B4h **Offset End:** B5h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved			
08	TXP	Transactions Pending: 0 = Controller has received all non-posted requests. 1 = Controller has issued non-posted requests which has not been completed.			RO
07 :01	Reserved	Reserved			
00		Initiate FLR — Used to initiate FLR transition. A write of ‘1’ indicates FLR transition. Since hardware must no t respond to any cycles till FLR completion the value read by software from this bit is ‘0’.			RW

8.1.5.16 Offset C0h: ATC—APM Trapping Control Register (SATA–B0:D31:F2)

Table 8-48. Offset C0h: ATC—APM Trapping Control Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** C0h **Offset End:** C0h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved			
03	SST	Secondary Slave Trap — Enables trapping and SMI# assertion on legacy I/O accesses to 170h-177h and 376h. The active device on the secondary interface must be device 1 for the trap and/or SMI# to occur.			RW
02	SPT	Secondary Master Trap — Enables trapping and SMI# assertion on legacy I/O accesses to 170h-177h and 376h. The active device on the secondary interface must be device 0 for the trap and/or SMI# to occur.			RW
01	PST	Primary Slave Trap — Enables trapping and SMI# assertion on legacy I/O accesses to 1F0h-1F7h and 3F6h. The active device on the primary interface must be device 1 for the trap and/or SMI# to occur.			RW
00	PMT	Primary Master Trap — Enables trapping and SMI# assertion on legacy I/O accesses to 1F0h-1F7h and 3F6h. The active device on the primary interface must be device 0 for the trap and/or SMI# to occur.			RW

8.1.5.17 Offset C4h: ATS—APM Trapping Status Register (SATA–B0:D31:F2)

Table 8-49. Offset C4h: ATS—APM Trapping Status Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** C4h **Offset End:** C4h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved			
03	SST	Secondary Slave Trap — Indicates that a trap occurred to the secondary slave device.			RWC
02	SPT	Secondary Master Trap — Indicates that a trap occurred to the secondary master device.			RWC
01	PST	Primary Slave Trap — Indicates that a trap occurred to the primary slave device.			RWC
00	PMT	Primary Master Trap — Indicates that a trap occurred to the primary master device.			RWC

8.1.5.18 Offset D0h: SP—Scratch Pad Register (SATA–B0:D31:F2)

Table 8-50. Offset D0h: SP—Scratch Pad Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** D0h **Offset End:** D0h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DT	Data — This is a read/write register that is available for software to use. No hardware action is taken on this register.			RW

8.1.5.19 Offset E0h: BFCS—BIST FIS Control/Status Register (SATA–B0:D31:F2)

Table 8-51. Offset E0h: BFCS—BIST FIS Control/Status Register (SATA–B0:D31:F2) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** E0h **Offset End:** E3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	P5BFI	Port 5 BIST FIS Initiate — When a rising edge is detected on this bit field, the PCH initiates a BIST FIS to the device on Port 5, using the parameters specified in this register and the data specified in BFTD1 and BFTD2. The BIST FIS will only be initiated if a device on Port 5 is present and ready (not partial/slumber state). After a BIST FIS is successfully completed, software must disable and re-enable the port using the PxE bits at offset 92h prior to attempting additional BIST FISs or to return the PCH to a normal operational mode. If the BIST FIS fails to complete, as indicated by the BFF bit in the register, then software can clear then set the P5BFI bit to initiate another BIST FIS. This can be retried until the BIST FIS eventually completes successfully			RW
14	P4BFI	Port 4 BIST FIS Initiate — When a rising edge is detected on this bit field, the PCH initiates a BIST FIS to the device on Port 4, using the parameters specified in this register and the data specified in BFTD1 and BFTD2. The BIST FIS will only be initiated if a device on Port 4 is present and ready (not partial/slumber state). After a BIST FIS is successfully completed, software must disable and re-enable the port using the PxE bits at offset 92h prior to attempting additional BIST FISs or to return the PCH to a normal operational mode. If the BIST FIS fails to complete, as indicated by the BFF bit in the register, then software can clear then set the P4BFI bit to initiate another BIST FIS. This can be retried until the BIST FIS eventually completes successfully			RW

Table 8-51. Offset E0h: BFCS—BIST FIS Control/Status Register (SATA–B0:D31:F2) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** E0h **Offset End:** E3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
13 :12	Reserved	Reserved			
11	BFS	BIST FIS Successful: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set any time a BIST FIS transmitted by PCH receives an R_OK completion status from the device. Note: This bit must be cleared by software prior to initiating a BIST FIS.			RWC
10	BFF	BIST FIS Failed: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set any time a BIST FIS transmitted by PCH receives an R_ERR completion status from the device. Note: This bit must be cleared by software prior to initiating a BIST FIS.			RWC
09 :08	Reserved	Reserved			
07 :02	BFP	BIST FIS Parameters — These 6 bits form the contents of the upper 6 bits of the BIST FIS Pattern Definition in any BIST FIS transmitted by the PCH. This field is not port specific — its contents will be used for any BIST FIS initiated on port 0, port 1, port 2 or port 3. The specific bit definitions are: Bit 7: T – Far End Transmit mode Bit 6: A – Align Bypass mode Bit 5: S – Bypass Scrambling Bit 4: L – Far End Retimed Loopback Bit 3: F – Far End Analog Loopback Bit 2: P – Primitive bit for use with Transmit mode			RW
01 :00	Reserved	Reserved			

8.1.5.20 Offset E4h: BFTD1—BIST FIS Transmit Data1 Register (SATA–B0:D31:F2)

Table 8-52. Offset E4h: FBTD1—BIST FIS Transmit Data1 Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** E4h **Offset End:** E7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIST FIS Transmit Data 1 — The data programmed into this register will form the contents of the second dword of any BIST FIS initiated by the PCH. This register is not port specific — its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the "T" bit of the BIST FIS is set to indicate "Far-End Transmit mode", this register's contents will be transmitted as the BIST FIS 2nd DW regardless of whether or not the "T" bit is indicated in the BFCS register (B0:D31:F2:E0h).			RW

8.1.5.21 Offset E8h: BFTD2—BIST FIS Transmit Data2 Register (SATA–B0:D31:F2)

Table 8-53. Offset E8h: BFTD2—BIST FIS Transmit Data2 Register (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** E8h **Offset End:** EBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIST FIS Transmit Data 2 — The data programmed into this register will form the contents of the third dword of any BIST FIS initiated by the PCH. This register is not port specific — its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the "T" bit of the BIST FIS is set to indicate "Far-End Transmit mode", this register's contents will be transmitted as the BIST FIS 3rd DW regardless of whether or not the "T" bit is indicated in the BFCS register (B0:D31:F2:E0h).			RW

8.1.6 SATA* Indexed Registers

The SATA* Indexed Registers are the registers used with the Offset 0Ah: SATA* Indexed Registers Index (SATA-B0:D31:F2) register.

Table 8-54. SATA* Indexed Registers

Offset Start	Offset End	Register ID - Description	Default Value
18h	1Bh	"Offset 18h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 457	00000000h
1Ch	1Fh	"Offset 1Ch: SATA* Indexed Registers Index (SATA* Test Mode Enable Register) (SATA–B0:D31:F2)" on page 457	00000000h
28h	2Bh	"Offset 28h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 458	0B000040h
3Eh	3Fh	"Offset 3Eh: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 458	04A4h
54h	57h	"Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 459	00AAAAAAh
64h	67h	"Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 459	84848484h
68h	69h	"Offset 68h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 460	8484h
78h	7Bh	"Offset 78h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 460	33220000h
84h	87h	"Offset 84h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 461	00000000h
88h	8Bh	"Offset 88h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 461	33332222h
8Ch	8Fh	"Offset 8Ch: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 462	55555555h
94h	95h	"Offset 94h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 462	5555h
A0h	A3h	"Offset A0h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 463	00000000h
A8h	ABh	"Offset A8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 463	04040404h
C4h	C7h	"Offset C4h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 464	04040404h
C8h	CBh	"Offset C8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)" on page 464	04040404h

8.1.6.1 Offset 18h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-55. Offset 18h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 001C7000h			

8.1.6.2 Offset 1Ch: SATA* Indexed Registers Index (SATA* Test Mode Enable Register) (SATA–B0:D31:F2)

Table 8-56. Offset 1Ch: SATA* Indexed Registers Index (SATA* Test Mode Enable Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	Reserved	Reserved			
18		SATA* Test Mode Enable Bit: 0 = Entrance to the PCH SATA* test modes are disabled. 1 = This bit allows entrance to the PCH SATA* test modes when set. This bit should only be used for system board testing.			RW
17 :00	Reserved	Reserved			

8.1.6.3 Offset 28h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-57. Offset 28h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 28h **Offset End:** 2Bh	
Size: 32 bit	**Default:** 0B000040h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 0A000033h.			

8.1.6.4 Offset 3Eh: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-58. Offset 3Eh: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 3Eh **Offset End:** 3Fh	
Size: 16 bit	**Default:** 04A4h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		BIOS programs this field to 0464h.			

8.1.6.5 Offset 54h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-59. Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 00AAAAAAh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :00		BIOS programs this field to 111111h.			

8.1.6.6 Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-60. Offset 64h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 84848484h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to CCCCCCCCh			

8.1.6.7 Offset 68h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-61. Offset 68h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 68h **Offset End:** 69h	
Size: 16 bit	**Default:** 8484h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		BIOS programs this field to CCCCh			

8.1.6.8 Offset 78h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-62. Offset 78h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 33220000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		BIOS programs this field to 99AAh			
15 :00	Reserved	Reserved			

8.1.6.9 Offset 84h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-63. Offset 84h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 001C7000h			

8.1.6.10 Offset 88h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-64. Offset 88h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 88h **Offset End:** 8Bh	
Size: 32 bit	**Default:** 33332222h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 9999AAAAh			

8.1.6.11 Offset 8Ch: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-65. Offset 8Ch: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 8Ch **Offset End:** 8Fh	
Size: 32 bit	**Default:** 55555555h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 55555555h			

8.1.6.12 Offset 94h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-66. Offset 94h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 94h **Offset End:** 95h	
Size: 16 bit	**Default:** 5555h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		BIOS programs this field to 5555h			

8.1.6.13 Offset A0h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-67. Offset A0h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 001C7000h			

8.1.6.14 Offset A8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-68. Offset A8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** A8h **Offset End:** ABh	
Size: 32 bit	**Default:** 04040404h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 0C0C0C0Ch			

8.1.6.15 Offset C4h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-69. Offset C4h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 04040404h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 0C0C0C0Ch			

8.1.6.16 Offset C8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Table 8-70. Offset C8h: SATA* Indexed Registers Index (SATA* Initialization Register) (SATA–B0:D31:F2)

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** C8h **Offset End:** CBh	
Size: 32 bit	**Default:** 04040404h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		BIOS programs this field to 0C0C0C0Ch			

8.2 Bus Master IDE I/O Registers (B0:D31:F2)

The bus master IDE function uses 16 bytes of I/O space, allocated via the LBAR register, located in Device 31:Function 2 Configuration space, offset 20h. All bus master IDE I/O space registers can be accessed as byte, word, or dword quantities. Reading reserved bits returns an indeterminate, inconsistent value, and writes to reserved bits have no affect (but should not be attempted). These registers are only used for legacy operation. Software must not use these registers when running AHCI. All I/O registers are reset by Function Level Reset. The description of the I/O registers is shown in the following table.

8.2.1 Bus Master IDE I/O Register Address Map

Table 8-71. Bus Master IDE I/O Register Address Map

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: Bus Master IDE Command Register Primary (B0:D31:F2)" on page 466	00h
02h	02h	"Offset 02h: Bus Master IDE Status Register Primary (B0:D31:F2)" on page 467	00h
04h	07h	"Offset 04h: Bus Master IDE Descriptor Table Pointer Register Primary (B0:D31:F2)" on page 468	00h
08h	08h	"Offset 08h: Bus Master IDE Command Register Secondary (B0:D31:F2)" on page 469	00h
0Ah	0Ah	"Offset 0Ah: Bus Master IDE Status Register Secondary (B0:D31:F2)" on page 470	00h
0Ch	0Fh	"Offset 0Ch: Bus Master IDE Descriptor Table Pointer Register Secondary (B0:D31:F2)" on page 471	00h
10h	10h	"Offset 10h: AHCI Index Register (B0:D31:F2)" on page 471	00000000h
14h	14h	"Offset 14h: AHCI Index Data Register (B0:D31:F2)" on page 472	00h

8.2.1.1 Offset 00h: Bus Master IDE Command Register Primary (B0:D31:F2)

Table 8-72. Offset 00h: Bus Master IDE Command Register Primary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved. Returns 0.			
03	R/WC	Read / Write Control — This bit sets the direction of the bus master transfer: This bit must NOT be changed when the bus master function is active. 0 = Memory reads 1 = Memory writes			RW
02 :01	Reserved	Reserved. Returns 0.			
00	START	Start/Stop Bus Master: 0 = All state information is lost when this bit is cleared. Master mode operation cannot be stopped and then resumed. If this bit is reset while bus master operation is still active (for example, the Bus Master IDE Active bit (B0:D31:F2:BAR + 02h, bit 0) of the Bus Master IDE Status register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded instead of being written to system memory. 1 = Enables bus master operation of the controller. Bus master operation does not actually start unless the Bus Master Enable bit (D31:F1:04h, bit 2) in PCI configuration space is also set. Bus master operation begins when this bit is detected changing from 0 to 1. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation can be halted by writing a 0 to this bit. Note: This bit is intended to be cleared by software after the data transfer is completed, as indicated by either the Bus Master IDE Active bit being cleared or the Interrupt bit of the Bus Master IDE Status register for that IDE channel being set, or both. Hardware does not clear this bit automatically. If this bit is cleared to 0 prior to the DMA data transfer being initiated by the drive in a device to memory data transfer, then the PCH will not send DMAT to terminate the data transfer. SW intervention (for example, sending SRST) is required to reset the interface in this condition.			RW

8.2.1.2 Offset 02h: Bus Master IDE Status Register Primary (B0:D31:F2)

Table 8-73. Offset 02h: Bus Master IDE Status Register Primary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	PRDIS	PRD Interrupt Status: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the host controller execution of a PRD that has its PRD_INT bit set.			RWC
06		Drive 1 DMA Capable: 0 = Not Capable. 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The PCH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.			RW
05		Drive 0 DMA Capable: 0 = Not Capable 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The PCH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.			RW
04 :03	Reserved	Reserved. Returns 0.			
02		Interrupt: 0 = Software clears this bit by writing a 1 to it. 1 = Set when a device FIS is received with the 'I' bit set, provided that software has not disabled interrupts via the IEN bit of the Device Control Register (see chapter 5 of the *Serial ATA Specification*, Revision 1.0a).			RWC
01		Error: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the controller encounters a target abort or master abort when transferring data on PCI.			RWC
00	ACT	Bus Master IDE Active: 0 = This bit is cleared by the PCH when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared by the PCH when the Start Bus Master bit (B0:D31:F2:BAR+ 00h, bit 0) is cleared in the Command register. When this bit is read as a 0, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted. 1 = Set by the PCH when the Start bit is written to the Command register.			RO

8.2.1.3 Offset 04h: Bus Master IDE Descriptor Table Pointer Register Primary (B0:D31:F2)

Table 8-74. Offset 04h: Bus Master IDE Descriptor Table Pointer Register Primary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address of Descriptor Table: The bits in this field correspond to bits [31:2] of the memory location of the Physical Region Descriptor (PRD). The Descriptor Table must be Dword-aligned. The Descriptor Table must not cross a 64-K boundary in memory.			RW
01 :00	Reserved	Reserved			

8.2.1.4 Offset 08h: Bus Master IDE Command Register Secondary (B0:D31:F2)

Table 8-75. Offset 08h: Bus Master IDE Command Register Secondary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved. Returns 0.			
03	R/WC	Read / Write Control — This bit sets the direction of the bus master transfer: This bit must NOT be changed when the bus master function is active. 0 = Memory reads 1 = Memory writes			RW
02 :01	Reserved	Reserved. Returns 0.			
00	START	Start/Stop Bus Master: 0 = All state information is lost when this bit is cleared. Master mode operation cannot be stopped and then resumed. If this bit is reset while bus master operation is still active (for example, the Bus Master IDE Active bit (B0:D31:F2:BAR + 02h, bit 0) of the Bus Master IDE Status register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded instead of being written to system memory. 1 = Enables bus master operation of the controller. Bus master operation does not actually start unless the Bus Master Enable bit (D31:F1:04h, bit 2) in PCI configuration space is also set. Bus master operation begins when this bit is detected changing from 0 to 1. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation can be halted by writing a 0 to this bit. Note: This bit is intended to be cleared by software after the data transfer is completed, as indicated by either the Bus Master IDE Active bit being cleared or the Interrupt bit of the Bus Master IDE Status register for that IDE channel being set, or both. Hardware does not clear this bit automatically. If this bit is cleared to 0 prior to the DMA data transfer being initiated by the drive in a device to memory data transfer, then the PCH will not send DMAT to terminate the data transfer. SW intervention (for example, sending SRST) is required to reset the interface in this condition.			RW

8.2.1.5 Offset 0Ah: Bus Master IDE Status Register Secondary (B0:D31:F2)

Table 8-76. Offset 0Ah: Bus Master IDE Status Register Secondary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	PRDIS	PRD Interrupt Status: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the host controller execution of a PRD that has its PRD_INT bit set.			RWC
06		Drive 1 DMA Capable: 0 = Not Capable. 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 1 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The PCH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.			RW
05		Drive 0 DMA Capable: 0 = Not Capable 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The PCH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.			RW
04 :03	Reserved	Reserved. Returns 0.			
02		Interrupt: 0 = Software clears this bit by writing a 1 to it. 1 = Set when a device FIS is received with the 'I' bit set, provided that software has not disabled interrupts via the IEN bit of the Device Control Register (see chapter 5 of the *Serial ATA Specification*, Revision 1.0a).			RWC
01		Error: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the controller encounters a target abort or master abort when transferring data on PCI.			RWC
00	ACT	Bus Master IDE Active: 0 = This bit is cleared by the PCH when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared by the PCH when the Start Bus Master bit (B0:D31:F2:BAR+ 00h, bit 0) is cleared in the Command register. When this bit is read as a 0, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted. 1 = Set by the PCH when the Start bit is written to the Command register.			RO

8.2.1.6 Offset 0Ch: Bus Master IDE Descriptor Table Pointer Register Secondary (B0:D31:F2)

Table 8-77. Offset 0Ch: Bus Master IDE Descriptor Table Pointer Register Secondary (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address of Descriptor Table: The bits in this field correspond to bits [31:2] of the memory location of the Physical Region Descriptor (PRD). The Descriptor Table must be Dword-aligned. The Descriptor Table must not cross a 64-K boundary in memory.			RW
01 :00	Reserved	Reserved			

8.2.1.7 Offset 10h: AHCI Index Register (B0:D31:F2)

Table 8-78. Offset 10h: AHCI Index Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 10h **Offset End:** 10h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :11	Reserved	Reserved			
10 :02	INDEX	Index — This Index register is used to select the Dword offset of the Memory Mapped AHCI register to be accessed. A Dword, Word or Byte access is specified by the active byte enables of the I/O access to the Data register.			RW
01 :00	Reserved	Reserved			

8.2.1.8 Offset 14h: AHCI Index Data Register (B0:D31:F2)

This register is available only when SCC is not 01h.

Table 8-79. Offset 14h: AHCI Index Data Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 14h **Offset End:** 14h	
Size: 32 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DTA	Data — This Data register is a "window" through which data is read or written to the AHCI memory mapped registers. A read or write to this Data register triggers a corresponding read or write to the memory mapped register pointed to by the Index register. The Index register must be setup prior to the read or write to this Data register. ***Note:*** A physical register is not actually implemented as the data is actually stored in the memory mapped registers. Since this is not a physical register, the "default" value is the same as the default value of the register pointed to by Index.			RW

8.3 Serial ATA Index/Data Pair Superset Registers

All of these I/O registers are in the core well. They are exposed only when SCC is 01h (for example, IDE programming interface).

These are Index/Data Pair registers that are used to access the SerialATA superset registers (SerialATA Status, SerialATA Control and SerialATA Error). The I/O space for these registers is allocated through SIDPBA. Locations with offset from 08h to 0Fh are reserved for future expansion. Software-write operations to the reserved locations will have no effect while software-read operations to the reserved locations will return 0.

8.3.1 Superset Registers

Table 8-80. Superset Registers

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset SIDPBA + 00h: Serial ATA Index (B0:D31:F2)" on page 473	00000000h
04h	07h	"Offset SIDPBA + 04h: Serial ATA Data (B0:D31:F2)" on page 474	00000000h

8.3.1.1 Offset SIDPBA + 00h: Serial ATA Index (B0:D31:F2)

Table 8-81. Offset SIDPBA + 00h: Serial ATA Index (B0:D31:F2)

Description:					
View: PCI	**BAR:** SIDPBA		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :08	PIDX	Port Index — This Index field is used to specify the port of the SATA* controller at which the port-specific SSTS, SCTL, and SERR registers are located. 00h = Primary Master (Port 0) 01h = Primary Slave (Port 2) 02h = Secondary Master (Port 1) 03h = Secondary Slave (Port 3) All other values are Reserved.			RW
07 :00	RIDX	Register Index — This index field is used to specify one out of three registers currently being indexed into. These three registers are the Serial ATA superset SStatus, SControl and SError memory registers and are port specific, hence for this SATA* controller, there are four sets of these registers. 00h = SSTS 01h = SCTL 02h = SERR All other values are Reserved.			RW

8.3.1.2 Offset 04h: Serial ATA Data (B0:D31:F2)

Table 8-82. Offset SIDPBA + 04h: Serial ATA Data (B0:D31:F2)

Description:					
View: PCI	**BAR:** SIDPBA		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DTA	Data — This Data register is a "window" through which data is read or written to from the register pointed to by the Serial ATA Index (SINDX) register above. A physical register is not implemented; the data is stored in the memory mapped registers. Since this is not a physical register, the "default" value is the same as the default value of the register pointed to by SINDX.RIDX field.			RW

8.4 AHCI Registers (B0:D31:F2)

Note: These registers are AHCI-specific and available when the PCH is properly configured. The Serial ATA Status, Control, and Error registers are special exceptions and may be accessed on all PCH components if properly configured.

The memory mapped registers within the SATA* controller exist in non-cacheable memory space. Additionally, locked accesses are not supported. If software attempts to perform locked transactions to the registers, indeterminate results may occur. Register accesses has a maximum size of 64-bits; 64-bit access must not cross an 8-byte alignment boundary. All memory registers are reset by Function Level Reset unless specified otherwise.

The registers are broken into two sections – generic host control and port control. The port control registers are the same for all ports, and there are as many registers banks as there are ports.

Table 8-83. AHCI Register Address Map

ABAR + Offset	Mnemonic	Register
00–1Fh	GHC	Generic Host Control
20h–FFh	—	Reserved
300h–37Fh	P4PCR	Port 4 port control registers
380h–3FFh	P5PCR	Port 5 port control registers

8.4.1 AHCI Generic Host Control Registers (B0:D31:F2)

Table 8-84. AHCI Generic Host Control Registers

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: Host Capabilities Register (B0:D31:F2)" on page 475	DE127F03h
04h	07h	"Offset 04h: Global PCH Control Register (B0:D31:F2)" on page 478	00000000h
08h	0Bh	"Offset 08h: Interrupt Status Register (B0:D31:F2)" on page 479	00000000h
OCh	OFh	"Offset 0Ch: Ports Implemented Register (B0:D31:F2)" on page 480	00000000h
10h	13h	"Offset 10h: Serial AHCI Version (B0:D31:F2)" on page 480	00010200h
14h	17h	"Offset 14h: Serial Command Completion Coalescing Control Register (B0:D31:F2)" on page 481	00000000h
18h	1Bh	"Offset 18h: Serial Command Completion Coalescing Ports Register (B0:D31:F2)" on page 482	00000000h
1Ch	1Fh	"Offset 1Ch: Serial Enclosure Management Location Register (B0:D31:F2)" on page 482	01600002h
20h	23h	"Offset 20h: Serial Enclosure Management Control Register (B0:D31:F2)" on page 483	07010000h
24h	24h	"Offset 24h: Serial Extended Host Capabilities (B0:D31:F2)" on page 484	0000000Xh
6Ch	6Ch	"Offset SIDPBA + 04h: Serial ATA Index (B0:D31:F2)" on page 485	00000001h
70h	73h	"Offset 70h: Serial AHCI Version (B0:D31:F2)" on page 485	00010000h
A0h	A3h	"Offset A0h: Serial Vendor Specific (B0:D31:F2)" on page 486	00000000h

8.4.1.1 Offset 00h: Host Capabilities Register (B0:D31:F2)

All bits in this register that are RWO are reset only by PLTRST#.

Table 8-85. Offset 00h: Host Capabilities Register (B0:D31:F2) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** DE127F03h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	S64A	Supports 64-bit Addressing — Indicates that the SATA* controller can access 64-bit data structures. The 32-bit upper bits of the port DMA Descriptor, the PRD Base, and each PRD entry are read/write.			RO
30	SCQA	Supports Command Queue Acceleration — Hardwired to 1 to indicate that the SATA* controller supports SATA* command queuing via the DMA Setup FIS. The PCH handles DMA Setup FISes natively, and can handle auto-activate optimization through that FIS.			RO
29	SSNTF	Supports SNotification Register — The PCH SATA* Controller does not support the SNotification register.			RO

Table 8-85. Offset 00h: Host Capabilities Register (B0:D31:F2) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** DE127F03h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
28	SIS	Supports Interlock Switch — Indicates whether the SATA* controller supports interlock switches on its ports for use in Hot Plug operations. This value is loaded by platform BIOS prior to OS initialization. If this bit is set, BIOS must also map the SATAGP pins to the SATA* controller through GPIO space.			RWO
27	SSS	Supports Staggered Spin-up — Indicates whether the SATA* controller supports staggered spin-up on its ports, for use in balancing power spikes. This value is loaded by platform BIOS prior to OS initialization. 0 = Staggered spin-up not supported. 1 = Staggered spin-up supported.			RWO
26	SALP	Supports Aggressive Link Power Management: 0 = Software shall treat the PxCMD.ALPE and PxCMD.ASP bits as reserved. 1 = The SATA* controller supports auto-generating link requests to the partial or slumber states when there are no commands to process.			RWO
25	SAL	Supports Activity LED — Indicates that the SATA* controller supports a single output pin (SATALED#) which indicates activity.			RO
24	SCLO	Supports Command List Override — When set to '1', indicates that the Controller supports the PxCMD.CLO bit and it's associated function. When cleared to '0', The Controller is not capable of clearing the BSY and DRQ bits in the Status register in order to issue a software reset if these bits are still set from a previous operation.			RWO
23 :20	ISS	Interface Speed Support — Indicates the maximum speed the SATA* controller can support on its ports. 2h =3.0 Gb/s.			RWO
19	SNZO	Supports Non-Zero DMA Offsets — Reserved, as per the AHCI Revision 1.2 specification			RO
18	SAM	Supports AHCI Mode Only — The SATA* controller may optionally support AHCI access mechanism only. 0 = SATA* controller supports both IDE and AHCI Modes 1 = SATA* controller supports AHCI Mode Only			RO
17 :16	Reserved	Reserved			
15	PMD	PIO Multiple DRQ Block — The SATA* controller supports PIO Multiple DRQ Command Block			RO
14	SSC	Slumber State Capable — The SATA* controller supports the slumber state.			RWO
13	PSC	Partial State Capable — The SATA* controller supports the partial state.			RWO
12 :08	NCS	Number of Command Slots — Hardwired to 1Fh to indicate support for 32 slots.			RO
07	CCCS	Command Completion Coalescing Supported: 0 = Command Completion Coalescing Not Supported 1 = Command Completion Coalescing Supported			RWO

Table 8-85. Offset 00h: Host Capabilities Register (B0:D31:F2) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** DE127F03h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	EMS	Enclosure Management Supported: 0 = Enclosure Management Not Supported 1 = Enclosure Management Supported			RWO
05	SXS	Supports External SATA*: 0 = External SATA* is not supported on any ports 1 = External SATA* is supported on one or more ports When set, SW can examine each SATA* port's Command Register (PxCMD) to determine which port is routed externally.			RWO
04 :00	NPS	Number of Ports — Indicates number of supported ports. The number of ports indicated in this field may be more than the number of ports indicated in the PI (ABAR + 0Ch) register.			RO

8.4.1.2 Offset 04h: Global PCH Control Register (B0:D31:F2)

Table 8-86. Offset 04h: Global PCH Control Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	AE	AHCI Enable: When set, indicates that an AHCI driver is loaded and the controller will be talked to via AHCI mechanisms. This can be used by a PCH that supports both legacy mechanisms (such as SFF-8038i) and AHCI to know when the controller will not be talked to as legacy. 0 = Software will communicate with the PCH using legacy mechanisms. 1 = Software will communicate with the PCH using AHCI. The PCH will not have to allow command processing via both AHCI and legacy mechanisms. Software shall set this bit to 1 before accessing other AHCI registers.			RW
30 :03	Reserved	Reserved			
02	MRSM	MSI Revert to Single Message: When set to '1' by hardware, indicates that the host controller requested more than one MSI vector but has reverted to using the first vector only. When this bit is cleared to '0', the Controller has not reverted to single MSI mode (for example, hardware is already in single MSI mode, software has allocated the number of messages requested, or hardware is sharing interrupt vectors if MC.MME < MC.MMC). "MC.MSIE = '1' (MSI is enabled) "MC.MMC > 0 (multiple messages requested) "MC.MME > 0 (more than one message allocated) "MC.MME!= MC.MMC (messages allocated not equal to number requested) When this bit is set to '1', single MSI mode operation is in use and software is responsible for clearing bits in the IS register to clear interrupts. This bit shall be cleared to '0' by hardware when any of the four conditions stated is false. This bit is also cleared to '0' when MC.MSIE = '1' and MC.MME = 0h. In this case, the hardware has been programmed to use single MSI mode, and is not "reverting" to that mode. For PCH, the Controller shall always revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. This bit is ignored when GHC.HR = 1.			RO
01	IE	Interrupt Enable: This global bit enables interrupts from the PCH. 0 = All interrupt sources from all ports are disabled. 1 = Interrupts are allowed from the AHCI controller.			RW
00	HR	Controller Reset: Resets the PCH AHCI controller. 0 = No effect 1 = When set by SW, this bit causes an internal reset of the PCH AHCI controller. All state machines that relate to data transfers and queuing return to an idle condition, and all ports are re-initialized via COMRESET. For further details, consult section 12.3.3 of the *Serial ATA Advanced Host Controller Interface* specification.			RW

8.4.1.3 Offset 08h: Interrupt Status Register (B0:D31:F2)

This register indicates which of the ports within the controller have an interrupt pending and require service.

Table 8-87. Offset 08h: Interrupt Status Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 08h **Offset End:** 0Bh		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved. Returns 0.			
08	CIPS	Coalescing Interrupt Pending Status: 0 = No interrupt pending. 1 = A command completion coalescing interrupt has been generated.			RWC
07 06	Reserved	Reserved			
05	IPS[5]	Interrupt Pending Status Port[5]: 0 = No interrupt pending. 1 = Port 5 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt.			RWC
04	IPS[4]	Interrupt Pending Status Port[4]: 0 = No interrupt pending. 1 = Port 4 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt.			RWC
03 :00	Reserved	Reserved			

8.4.1.4 Offset 0Ch: Ports Implemented Register (B0:D31:F2)

This register indicates which ports are exposed to the PCH. It is loaded by platform BIOS. It indicates which ports that the device supports are available for software to use. For ports that are not available, software must not read or write to registers within that port.

Table 8-88. Offset 0Ch: Ports Implemented Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31 :F2		**Offset Start:** OCh **Offset End:** OFh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :06	Reserved	Reserved			
05	PI5	Ports Implemented Port 5: 0 = The port is not implemented. 1 = The port is implemented. This bit is read-only '0' if MAP.SC = '0' or SCC = '01h'.			RWO
04	PI4	Ports Implemented Port 4: 0 = The port is not implemented. 1 = The port is implemented. This bit is read-only '0' if MAP.SC = '0' or SCC = '01h'.			RWO
03 :00	Reserved	Reserved			

8.4.1.5 Offset 10h: Serial AHCI Version (B0:D31:F2)

This register indicates the major and minor version of the AHCI specification. It is BCD encoded. The upper two bytes represent the major version number, and the lower two bytes represent the minor version number. Example: Version 3.12 would be represented as 00030102h. The current version of the specification is 1.20 (00010200h).

Table 8-89. Offset 10h: Serial AHCI Version (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31 :F2		**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00010200h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	MJR	Major Version Number — Indicates the major version is 1			RO
15 :00	MNR	Minor Version Number — Indicates the minor version is 20.			RO

8.4.1.6 Offset 14h: Serial Command Completion Coalescing Control Register (B0:D31:F2)

This register is used to configure the command coalescing feature. This register is reserved if command coalescing is not supported (CAP_CCCS = '0').

Table 8-90. Offset 14h: Serial Command Completion Coalescing Control Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	TV	Timeout Value — The timeout value is specified in 10 microsecond intervals. hbaCCC_Timer is loaded with this timeout value. hbaCCC_Timer is only decremented when commands are outstanding on the selected ports. The Controller will signal a CCC interrupt when hbaCCC_Timer has decremented to '0'. The hbaCCC_Timer is reset to the timeout value on the assertion of each CCC interrupt. A timeout value of 0 is invalid.			RW
15 :08	CC	Command Completions — Specifies the number of command completions that are necessary to cause a CCC interrupt. The Controller has an internal command completion counter, hbaCCC_CommandsComplete. hbaCCC_CommandsComplete is incremented by one each time a selected port has a command completion. When hbaCCC_CommandsComplete is equal to the command completions value, a CCC interrupt is signaled. The internal command completion counter is reset to '0' on the assertion of each CCC interrupt.			RW
07 :03	INT	Interrupt — Specifies the interrupt used by the CCC feature. This interrupt must be marked as unused in the AHCI Ports Implemented memory register by the corresponding bit being set to '0'. Thus, the CCC_interrupt corresponds to the interrupt for an unimplemented port on the controller. When a CCC interrupt occurs, the IS[INT] bit shall be asserted to '1' regardless of whether PIRQ interrupt or MSI is used. In MSI, CC interrupt may share an interrupt vector with other ports. For example, if the number of message allocated is 4, then CCC interrupt share interrupt vector 3 along with port 3, 4, and 5 but IS[6] shall get set.			RO
02 :01	Reserved	Reserved			
00	EN	Enable: 0 = The command completion coalescing feature is disabled and no CCC interrupts are generated 1 = The command completion coalescing feature is enabled and CCC interrupts may be generated based on timeout or command completion conditions. Software shall only change the contents of the TV and CC fields when EN is cleared to '0'. On transition of this bit from '0' to '1', any updated values for the TV and CC fields shall take effect.			RW

8.4.1.7 Offset 18h: Serial Command Completion Coalescing Ports Register (B0:D31:F2)

This register is used to specify the ports that are coalesced as part of the CCC feature when CCC_CTL.EN = '1'. This register is reserved if command coalescing is not supported (CAP_CCCS = '0').

Table 8-91. Offset 18h: Serial Command Completion Coalescing Ports Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 18h **Offset End:** 1Bh		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRT	Ports: 0 = The port is not part of the command completion coalescing feature. 1 = The corresponding port is part of the command completion coalescing feature. Bits set to '1' in this register must also have the corresponding bit set to '1' in the Ports Implemented register. Bits set to '1' in this register must also have the corresponding bit set to '1' in the Ports Implemented register. An updated value for this field shall take effect within one timer increment (1 millisecond).			RW

8.4.1.8 Offset 1Ch: Serial Enclosure Management Location Register (B0:D31:F2)

This register identifies the location and size of the enclosure management message buffer. This register is reserved if enclosure management is not supported (for example, CAP.EMS = 0).

Table 8-92. Offset 1Ch: Serial Enclosure Management Location Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 1Ch **Offset End:** 1Fh		
Size: 32 bit	**Default:** 01600002h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	OFST	Offset — The offset of the message buffer in Dwords from the beginning of the ABAR.			RO
15 :00	SZ	Buffer Size — Specifies the size of the transmit message buffer area in Dwords. The PCH SATA* controller only supports transmit buffer. A value of '0' is invalid.			RO

8.4.1.9 Offset 20h: Serial Enclosure Management Control Register (B0:D31:F2)

This register is used to control and obtain status for the enclosure management interface. This register includes information on the attributes of the implementation, enclosure management messages supported, the status of the interface, whether any message are pending, and is used to initiate sending messages. This register is reserved if enclosure management is not supported (CAP_EMS = '0').

Table 8-93. Offset 20h: Serial Enclosure Management Control Register (B0:D31:F2) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 20h **Offset End:** 23h		
Size: 32 bit	**Default:** 07010000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved			
26	ATTR.ALHD	Activity LED Hardware Driven: 1 = The SATA* controller drives the activity LED for the LED message type in hardware and does not utilize software for this LED. The host controller does not begin transmitting the hardware based activity signal until after software has written CTL.TM=1 after a reset condition.			RWO
25	ATTR.XMT	Transmit Only: 0 = The SATA* controller supports transmitting and receiving messages. 1 = The SATA* controller only supports transmitting messages and does not support receiving messages.			RO
24	ATTR.SMB	Single Message Buffer: 0 = There are separate receive and transmit buffers such that unsolicited messages could be supported. 1 = The SATA* controller has one message buffer that is shared for messages to transmit and messages received. Unsolicited receive messages are not supported and it is software's responsibility to manage access to this buffer.			RO
23 :20	Reserved	Reserved			
19	SUPP.SGPIO	SGPIO Enclosure Management Messages: 1 = The SATA* controller supports the SGPIO register interface message type.			RO
18	SUPP.SES2	SES-2 Enclosure Management Messages: 1 = The SATA* controller supports the SES-2 message type.			RO
17	SUPP.SAFTE	SAF-TE Enclosure Management Messages: 1 = The SATA* controller supports the SAF-TE message type.			RO
16	SUPP.LED	LED Message Types: 1 = The SATA* controller supports the LED message type.			RO
15 :10	Reserved	Reserved			

Table 8-93. Offset 20h: Serial Enclosure Management Control Register (B0:D31:F2) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 07010000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	RST	Reset: 1 = A write of '0' to this bit by software will have no effect. 0 = When set by software, The SATA* controller shall reset all enclosure management message logic and take all appropriate reset actions to ensure messages can be transmitted / received after the reset. After the SATA* controller completes the reset operation, the SATA* controller shall set the value to '0'.			RW
31 :08	CTL.TM	Transmit Message: 0 = A write of '0' to this bit by software will have no effect. 1 = When set by software, The SATA* controller shall transmit the message contained in the message buffer. When the message is completely sent, the SATA* controller shall set the value to '0'. Software shall not change the contents of the message buffer while CTL.TM is set to '1'.			RW
07 :01	Reserved	Reserved			
00	STS.MR	Message Received — RO. Message Received is not supported in the PCH.			RO

8.4.1.10 Offset 24h: Serial Extended Host Capabilities (B0:D31:F2)

Table 8-94. Offset 24h: Serial Extended Host Capabilities (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 24h **Offset End:** 24h	
Size: 32 bit	**Default:** 0000000Xh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Reserved	Reserved			

8.4.1.11 Offset 6Ch: Serial NVMHCI Ports Implemented (B0:D31:F2)

Table 8-95. Offset SIDPBA + 04h: Serial ATA Index (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 6Ch **Offset End:** 6Ch	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Reserved	Reserved. Returns 0.			

8.4.1.12 Offset 70h: Serial AHCI Version (B0:D31:F2)

This register indicates the major and minor versions of the NVMHCI specification. It is BCD encoded. The upper two bytes represent the major version number and the lower two bytes represent the minor version number. Example: Version 3.12 would be represented as 00030102h. The current version of the specification is 1.0 (00010000h).

Table 8-96. Offset 70h: Serial AHCI Version (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00010000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	MJR	Major Version Number — Indicates the major version is 1			RO
15 :00	MNR	Minor Version Number — Indicates the minor version is 0.			RO

8.4.1.13 Offset A0h: Serial Vendor Specific (B0:D31:F2)

Table 8-97. Offset A0h: Serial Vendor Specific (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved			
00	SLPD	Supports Low Power Device Detection — Indicates whether SATA* power management and device hot (un)plug is supported. 0 = Not supported. 1 = Supported.			RWO

8.4.2 Port Registers (B0:D31:F2)

Ports not available will result in the corresponding Port DMA register space being reserved. The controller shall ignore writes to the reserved space on write cycles and shall return '0' on read cycle accesses to the reserved location.

There is a set of Port Registers for each Port 4 and 5.

Port 4 registers begin at ABAR + 300h

Port 5 registers begin at ABAR + 380h

Table 8-98. Port [4:5] DMA Register Address Map

Offset Start	Offset End	Register ID - Description	Default Value
300h at 80h	303h at 80h	"Offset 300h: Port [4:5] Command List Base Address Register (B0:D31:F2)" on page 488	Undefined
304h at 80h	307h at 80h	"Offset 304: Port [4:5] Command List Base Address Upper 32-Bits Register (B0:D31:F2)" on page 488	Undefined
308h + 80h	30Bh + 80h	"Offset 308h: Port [4:5] FIS Base Address Register (B0:D31:F2)" on page 489	Undefined
30Ch + 80h	30Fh + 80h	"Offset 30Ch: Port [4:5] FIS Base Address Upper 32-Bits Register (B0:D31:F2)" on page 489	Undefined
310h + 80h	313h + 80h	"Offset 310h: Port [4:5] Interrupt Status Register (B0:D31:F2)" on page 490	0000000h
314h + 80h	317h + 80h	"Offset 314h: Port [4:5] Interrupt Enable Register (B0:D31:F2)" on page 492	0000000h
318h + 80h	31Bh + 80h	"Offset 318h: Port [4:5] Command Register (B0:D31:F2)" on page 494	0000w00wh
320h + 80h	323h + 80h	"Offset 320h: Port [4:5] Task File Data Register (B0:D31:F2)" on page 497	0000007Fh
324h + 80h	327h + 80h	"Offset 324h: Port [4:5] Signature Register (B0:D31:F2)" on page 498	FFFFFFFFh
328h + 80h	32Bh + 80h	"Offset 328h: Port [4:5] Serial ATA Status Register (B0:D31:F2)" on page 499	00000000h
32Ch + 80h	32Fh + 80h	"Offset 32Ch: Port [4:5] Serial ATA Control Register (B0:D31:F2)" on page 501	000000004h
330h + 80h	333h + 80h	"Offset 330h: Port [4:5] Serial ATA Error Register (B0:D31:F2)" on page 503	00000000h
334h + 80h	337h + 80h	"Offset 334h: Port [4:5] Serial ATA Active (B0:D31:F2)" on page 504	00000000h
338h + 80h	33Bh + 80h	"Offset 338h: Port [4:5] Command Issue Register (B0:D31:F2)" on page 505	00000000h

Note: Port 4 offset starts at 300h. Port 5 offset starts at 380h.

8.4.2.1 Offset 300h: Port [4:5] Command List Base Address Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 300h
Port 5: ABAR + 380h

Table 8-99. Offset 300h: Port [4:5] Command List Base Address Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 300h at 80h **Offset End:** 303h at 80h	
Size: 32 bit	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	CLB	Command List Base Address — Indicates the 32-bit base for the command list for this port. This base is used when fetching commands to execute. The structure pointed to by this address range is 1 KB in length. This address must be 1-KB aligned as indicated by bits 31:10 being read/write. These bits are not reset on a Controller reset.			RW
09 :00	Reserved	Reserved			

8.4.2.2 Offset 304h: Port [4:5] Command List Base Address Upper 32-Bits Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 304h
Port 5: ABAR + 384h

Table 8-100. Offset 304: Port [4:5] Command List Base Address Upper 32-Bits Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 304h at 80h **Offset End:** 307h at 80h	
Size: 32 bit	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CLBU	Command List Base Address Upper — Indicates the upper 32-bits for the command list base address for this port. This base is used when fetching commands to execute. These bits are not reset on a Controller reset.			RW

8.4.2.3 Offset 308h: Port [4:5] FIS Base Address Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 308h
Port 5: ABAR + 388h

Table 8-101. Offset 308h: Port [4:5] FIS Base Address Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 308h + 80h **Offset End:** 30Bh + 80h	
Size: 32 bit	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	FB	FIS Base Address — Indicates the 32-bit base for received FISes. The structure pointed to by this address range is 256 bytes in length. This address must be 256-byte aligned, as indicated by bits 31:3 being read/write. These bits are not reset on a Controller reset.			RW
07 :00	Reserved	Reserved			

8.4.2.4 Offset 30Ch: Port [4:5] FIS Base Address Upper 32-Bits Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 30Ch
Port 5: ABAR + 38Ch

Table 8-102. Offset 30Ch: Port [4:5] FIS Base Address Upper 32-Bits Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 30Ch + 80h **Offset End:** 30Fh + 80h	
Size: 32 bit	**Default:** Undefined			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CLBU	Command List Base Address Upper — Indicates the upper 32-bits for the received FIS base for this port. These bits are not reset on a Controller reset.			RW

8.4.2.5 Offset 310h: Port [4:5] Interrupt Status Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 310h
Port 5: ABAR + 390h

Table 8-103. Offset 310h: Port [4:5] Interrupt Status Register (B0:D31:F2) (Sheet 1 of 2)

Description:						
View: PCI		**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 310h + 80h **Offset End:** 313h + 80h	
Size: 32 bit		**Default:** 0000000h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	31	CPDS	Cold Port Detect Status — Cold presence detect is not supported.			RO
	30	TFES	Task File Error Status — This bit is set whenever the status register is updated by the device and the error bit (PxTFD.bit 0) is set.			RWC
	29	HBFS	Host Bus Fatal Error Status — Indicates that the PCH encountered an error that it cannot recover from due to a bad software pointer. In PCI, such an indication would be a target or master abort.			RWC
	28	HBDS	Host Bus Data Error Status — Indicates that the PCH encountered a data error (uncorrectable ECC / parity) when reading from or writing to system memory.			RWC
	27	IFS	Interface Fatal Error Status — Indicates that the PCH encountered an error on the SATA* interface which caused the transfer to stop.			RWC
	26	INFS	Interface Non-fatal Error Status — Indicates that the PCH encountered an error on the SATA* interface but was able to continue operation.			RWC
	25	Reserved	Reserved			
	24	OFS	Overflow Status — indicates that the PCH received more bytes from a device than was specified in the PRD table for the command.			RWC
	23	Reserved	Reserved			
	22	PRCS	PhyRdy Change Status — When set to one indicates the internal PhyRdy signal changed state. This bit reflects the state of PxSERR.DIAG.N. Unlike most of the other bits in the register, this bit is RO and is only cleared when PxSERR.DIAG.N is cleared. The internal PhyRdy signal also transitions when the port interface enters partial or slumber power management states. Partial and slumber must be disabled when Surprise Removal Notification is desired, otherwise the power management state transitions will appear as false insertion and removal events.			RO
21	:08	Reserved	Reserved			
	07	DIS	Device Interlock Status — When set, indicates that a platform interlock switch has been opened or closed, which may lead to a change in the connection state of the device. This bit is only valid in systems that support an interlock switch (CAP.SIS [ABAR+00:bit 28] set). For systems that do not support an interlock switch, this bit will always be 0.			RWC

Table 8-103. Offset 310h: Port [4:5] Interrupt Status Register (B0:D31:F2) (Sheet 2 of 2)

Description:						
View: PCI		**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 310h + 80h **Offset End:** 313h + 80h	
Size: 32 bit		**Default:** 0000000h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	06	PCS	Port Connect Change Status — This bit reflects the state of PxSERR.DIAG.X. (ABAR+130h/1D0h/230h/2D0h, bit 26) Unlike other bits in this register, this bit is only cleared when PxSERR.DIAG.X is cleared. 0 = No change in Current Connect Status. 1 = Change in Current Connect Status.			RO
	05	DPS	Descriptor Processed — A PRD with the I bit set has transferred all its data.			RWC
	04	UFS	Unknown FIS Interrupt — When set to '1' indicates that an unknown FIS was received and has been copied into system memory. This bit is cleared to '0' by software clearing the PxSERR.DIAG.F bit to '0'. This bit does not directly reflect the PxSERR.DIAG.F bit. PxSERR.DIAG.F is set immediately when an unknown FIS is detected, whereas this bit is set when the FIS is posted to memory. Software should wait to act on an unknown FIS until this bit is set to '1' or the two bits may become out of sync.			RO
	03	SDBS	Set Device Bits Interrupt — A Set Device Bits FIS has been received with the I bit set and has been copied into system memory.			RWC
	02	DSS	DMA Setup FIS Interrupt — A DMA Setup FIS has been received with the I bit set and has been copied into system memory.			RWC
	01	PSS	PIO Setup FIS Interrupt — A PIO Setup FIS has been received with the I bit set, it has been copied into system memory, and the data related to that FIS has been transferred.			RWC
	00	DHRS	Device to Host Register FIS Interrupt — A D2H Register FIS has been received with the I bit set, and has been copied into system memory.			RWC

8.4.2.6 Offset 314h: Port [4:5] Interrupt Enable Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 314h
Port 5: ABAR + 394h

This register enables and disables the reporting of the corresponding interrupt to system software. When a bit is set (‘1’) and the corresponding interrupt condition is active, then an interrupt is generated. Interrupt sources that are disabled (‘0’) are still reflected in the status registers.

Table 8-104. Offset 314h: Port [4:5] Interrupt Enable Register (B0:D31:F2) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 314h + 80h **Offset End:** 317h + 80h	
Size: 32 bit	**Default:** 0000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	CPDE	Cold Presence Detect Enable — Cold Presence Detect is not supported.				RO
30	TFEE	Task File Error Enable — When set, and GHC.IE and PxTFD.STS.ERR (due to a reception of the error register from a received FIS) are set, the PCH will generate an interrupt.				RW
29	HBFE	Host Bus Fatal Error Enable — When set, and GHC.IE and PxS.HBFS are set, the PCH will generate an interrupt.				RW
28	HBDE	Host Bus Data Error Enable — When set, and GHC.IE and PxS.HBDS are set, the PCH will generate an interrupt.				RW
27	HBDE	Host Bus Data Error Enable — When set, GHC.IE is set, and PxIS.HBDS is set, the PCH will generate an interrupt.				RW
26	INFE	Interface Non-fatal Error Enable — When set, GHC.IE is set, and PxIS.INFS is set, the PCH will generate an interrupt.				RW
25	Reserved	Reserved				
24	OFE	Overflow Error Enable — When set, and GHC.IE and PxS.OFS are set, the PCH will generate an interrupt.				RW
23	Reserved	Reserved				
22	PRCE	PhyRdy Change Interrupt Enable — When set, and GHC.IE is set, and PxIS.PRCS is set, the PCH shall generate an interrupt.				RW
21:08	Reserved	Reserved				
07	DIE	Device Interlock Enable — When set, and PxIS.DIS is set, the PCH will generate an interrupt. For systems that do not support an interlock switch, this bit shall be a read-only 0.				RW
06	PCE	Port Change Interrupt Enable — When set, and GHC.IE and PxS.PCS are set, the PCH will generate an interrupt.				RW
05	DPE	Descriptor Processed Interrupt Enable — When set, and GHC.IE and PxS.DPS are set, the PCH will generate an interrupt				RW
04	UFIE	Unknown FIS Interrupt Enable — When set, and GHC.IE is set and an unknown FIS is received, the PCH will generate this interrupt.				RW

Table 8-104. Offset 314h: Port [4:5] Interrupt Enable Register (B0:D31:F2) (Sheet 2 of 2)

Description:							
View: PCI		**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 314h + 80h **Offset End:** 317h + 80h	
Size: 32 bit		**Default:** 0000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
	03	SDBE	Set Device Bits FIS Interrupt Enable — When set, and GHC.IE and PxS.SDBS are set, the PCH will generate an interrupt.				RW
	02	DSE	DMA Setup FIS Interrupt Enable — When set, and GHC.IE and PxS.DSS are set, the PCH will generate an interrupt.				RW
	01	PSE	PIO Setup FIS Interrupt Enable — When set, and GHC.IE and PxS.PSS are set, the PCH will generate an interrupt.				RW
	00	DHRE	Device to Host Register FIS Interrupt Enable — When set, and GHC.IE and PxS.DHRS are set, the PCH will generate an interrupt.				RW

8.4.2.7 Offset 318h: Port [4:5] Command Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 318h
Port 5: ABAR + 398h

Table 8-105. Offset 318h: Port [4:5] Command Register (B0:D31:F2) (Sheet 1 of 3)

Description:							
View: PCI		**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 318h + 80h **Offset End:** 31Bh + 80h	
Size: 32 bit		**Default:** 0000w00wh				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:28	ICC	Interface Communication Control — This is a four bit field which can be used to control reset and power states of the interface. Writes to this field will cause actions on the interface, either as primitives or an OOB sequence, and the resulting status of the interface will be reported in the PxSSTS register (Address offset Port 4: ABAR+224h, Port 5: ABAR+2A4h). **Value** / **Definition** Fh–7h: Reserved 6h: Slumber: This will cause the PCH to request a transition of the interface to the slumber state. The SATA* device may reject the request and the interface will remain in its current state 5h–3h: Reserved 2h: Partial: This will cause the PCH to request a transition of the interface to the partial state. The SATA* device may reject the request and the interface will remain in its current state. 1h: Active: This will cause the PCH to request a transition of the interface into the active 0h: No-Op / Idle: When software reads this value, it indicates the PCH is not in the process of changing the interface state or sending a device reset, and a new link command may be issued. When system software writes a non-reserved value other than No-Op (0h), the PCH will perform the action and update this field back to Idle (0h). If software writes to this field to change the state to a state the link is already in (for example, interface is in the active state and a request is made to go to the active state), the PCH will take no action and return this field to Idle. When the ALPE bit (bit 26) is set, then this register should not be set to 02h or 06h.			RW
	27	ASP	Aggressive Slumber / Partial — When set, and the ALPE bit (bit 26) is set, the PCH shall aggressively enter the slumber state when it clears the PxCI register and the PxSACT register is cleared. When cleared, and the ALPE bit is set, the PCH will aggressively enter the partial state when it clears the PxCI register and the PxSACT register is cleared. If CAP.SALP is cleared to '0', software shall treat this bit as reserved.				RW
	26	ALPE	Aggressive Link Power Management Enable — When set, the PCH will aggressively enter a lower link power state (partial or slumber) based upon the setting of the ASP bit (bit 27).				RW
	25	HBDE	Drive LED on ATAPI Enable — When set, the PCH will drive the LED pin active for ATAPI commands (PxCLB[CHz.A] set) in addition to ATA commands. When cleared, the PCH will only drive the LED pin active for ATA commands.				RW

Table 8-105. Offset 318h: Port [4:5] Command Register (B0:D31:F2) (Sheet 2 of 3)

Description:						
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 318h + 80h **Offset End:** 31Bh + 80h	
Size: 32 bit	**Default:** 0000w00wh				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
24	ATAPI	Device is ATAPI — When set, the connected device is an ATAPI device. This bit is used by the PCH to control whether or not to generate the desktop LED when commands are active.				RW
23	Reserved	Reserved				
22	Reserved	Reserved				
21	ESP	External SATA* Port: 0 = This port supports internal SATA* devices only. 1 = This port will be used with an external SATA* device. This bit is not reset by Function Level Reset.				RWO
20	Reserved	Reserved				
19	ISP	Interlock Switch Attached to Port — When interlock switches are supported in the platform (CAP.SIS [ABAR+00h:bit 28] set), this indicates whether this particular port has an interlock switch attached. This bit can be used by system software to enable such features as aggressive power management, as disconnects can always be detected regardless of PHY state with an interlock switch. When this bit is set, it is expected that HPCP (bit 18) in this register is also set. The PCH takes no action on the state of this bit – it is for system software only. For example, if this bit is cleared, and an interlock switch toggles, the PCH still treats it as a proper interlock switch event. This bit is not reset on a Controller reset or by a Function Level Reset.				RWO
18:16	Reserved	Reserved				
15	CR	Controller Running — When this bit is set, the DMA engines for a port are running. See section 5.2.2 of the *Serial ATA AHCI Specification* for details on when this bit is set and cleared by the PCH.				RO
14	FR	FIS Receive Running — When set, the FIS Receive DMA engine for the port is running. See section 12.2.2 of the *Serial ATA AHCI Specification* for details on when this bit is set and cleared by the PCH.				RO
13	ISS	Interlock Switch State — For systems that support interlock switches (via CAP.SIS [ABAR+00h:bit 28]), if an interlock switch exists on this port (via ISP in this register), this bit indicates the current state of the interlock switch. A 0 indicates the switch is closed, and a 1 indicates the switch is opened. For systems that do not support interlock switches, or if an interlock switch is not attached to this port, this bit reports 0.				RO
12:08	CSS	Current Command Slot — Indicates the current command slot the PCH is processing. This field is valid when the ST bit is set in this register, and is constantly updated by the PCH. This field can be updated as soon as the PCH recognizes an active command slot, or at some point soon after when it begins processing the command. This field is used by software to determine the current command issue location of the PCH. In queued mode, software shall not use this field, as its value does not represent the current command being executed. Software shall only use PxCI and PxSACT when running queued commands.				RO

Table 8-105. Offset 318h: Port [4:5] Command Register (B0:D31:F2) (Sheet 3 of 3)

Description:						
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 318h + 80h **Offset End:** 31Bh + 80h	
Size: 32 bit	**Default:** 0000w00wh				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07:05	Reserved	Reserved				
04	FRE	FIS Receive Enable — When set, the PCH may post received FISes into the FIS receive area pointed to by PxFB (ABAR+108h/188h/208h/288h) and PxFBU (ABAR+10Ch/18Ch/20Ch/28Ch). When cleared, received FISes are not accepted by the PCH, except for the first D2H (device-to-host) register FIS after the initialization sequence. System software must not set this bit until PxFB (PxFBU) have been programmed with a valid pointer to the FIS receive area, and if software wishes to move the base, this bit must first be cleared, and software must wait for the FR bit (bit 14) in this register to be cleared.				RW
03	CLO	Command List Override — Setting this bit to '1' causes PxTFD.STS.BSY and PxTFD.STS.DRQ to be cleared to '0'. This allows a software reset to be transmitted to the device regardless of whether the BSY and DRQ bits are still set in the PxTFD.STS register. The Controller sets this bit to '0' when PxTFD.STS.BSY and PxTFD.STS.DRQ have been cleared to '0'. A write to this register with a value of '0' has no effect. This bit shall only be set to '1' immediately prior to setting the PxCMD.ST bit to '1' from a previous value of '0'. Setting this bit to '1' at any other time is not supported and will result in indeterminate behavior. Software must wait for CLO to be cleared to '0' before setting PxCMD.ST to '1'.				RW
02	POD	Power On Device — Cold presence detect not supported. Defaults to 1.				RO
01	SUD	Spin-Up Device — This bit is R/W and defaults to 0 for systems that support staggered spin-up (R/W when CAP.SSS (ABAR+00h:bit 27) is 1). Bit is RO 1 for systems that do not support staggered spin-up (when CAP.SSS is 0). 0 = No action. 1 = On an edge detect from 0 to 1, the PCH starts a COMRESET initialization sequence to the device. Clearing this bit to '0' does not cause any OOB signal to be sent on the interface. When this bit is cleared to '0' and PxSCTL.DET=0h, the Controller will enter listen mode.				RW/RO
00	ST	Start — When set, the PCH may process the command list. When cleared, the PCH may not process the command list. Whenever this bit is changed from a 0 to a 1, the PCH starts processing the command list at entry 0. Whenever this bit is changed from a 1 to a 0, the PxCI register is cleared by the PCH upon the PCH putting the controller into an idle state. See section 12.2.1 of the Serial ATA AHCI Specification for important restrictions on when ST can be set to 1.				RW

8.4.2.8 Offset 320h: Port [4:5] Task File Data Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 320h
Port 5: ABAR + 3A0h

This is a 32-bit register that copies specific fields of the task file when FISes are received. The FISes that contain this information are: D2H Register FIS,PIO Setup FIS and Set Device Bits FIS.

Table 8-106. Offset 320h: Port [4:5] Task File Data Register (B0:D31:F2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR: ABAR</td><td colspan="2">Bus:Device:Function: B0:D31 :F2</td><td>Offset Start: 320h + 80h
Offset End: 323h + 80h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: 0000007Fh</td><td colspan="2"></td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :16</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>15 :08</td><td>ERR</td><td>Error — Contains the latest copy of the task file error register.</td><td></td><td></td><td>RO</td></tr>
<tr><td>07 :00</td><td>STS</td><td>Status — Contains the latest copy of the task file status register. Fields of note in this register that affect AHCI.
<table><tr><th>Bit</th><th>Field</th><th>Definition</th></tr><tr><td>7</td><td>BSY</td><td>Indicates the interface is busy</td></tr><tr><td>6:4</td><td>N/A</td><td>Not applicable</td></tr><tr><td>3</td><td>DRQ</td><td>Indicates a data transfer is requested</td></tr><tr><td>2:1</td><td>N/A</td><td>Not applicable</td></tr><tr><td>0</td><td>ERR</td><td>Indicates an error during the transfer</td></tr></table></td><td></td><td></td><td>RO</td></tr>
</table>

8.4.2.9 Offset 324h: Port [4:5] Signature Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 324h
Port 5: ABAR + 3A4h

This is a 32-bit register which contains the initial signature of an attached device when the first D2H Register FIS is received from that device. It is updated once after a reset sequence.

Table 8-107. Offset 324h: Port [4:5] Signature Register (B0:D31:F2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: ABAR</td><td>Bus:Device:Function: B0:D31:F2</td><td>Offset Start: 324h + 80h
Offset End: 327h + 80h</td></tr>
<tr><td colspan="2">Size: 32 bit</td><td colspan="2">Default: FFFFFFFFh</td><td></td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :00</td><td>SIG</td><td>Signature — Contains the signature received from a device on the first D2H register FIS. The bit order is as follows:
<table><tr><th>Bit</th><th>Field</th></tr><tr><td>31:24</td><td>LBA High Register</td></tr><tr><td>23:16</td><td>LBA Mid Register</td></tr><tr><td>15:8</td><td>LBA Low Register</td></tr><tr><td>7:0</td><td>Sector Count Register</td></tr></table></td><td></td><td></td><td>RO</td></tr>
</table>

8.4.2.10 Offset 328h: Port [4:5] Serial ATA Status Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 328h
Port 5: ABAR + 3A8h

This is a 32-bit register that conveys the current state of the interface and host. The PCH updates it continuously and asynchronously. When the PCH transmits a COMRESET to the device, this register is updated to its reset values.

Table 8-108. Offset 328h: Port [4:5] Serial ATA Status Register (B0:D31:F2) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 328h + 80h **Offset End:** 32Bh + 80h		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11 :08	IPM	Interface Power Management — Indicates the current interface state: **Value** / **Description** 0h — Device not present or communication not established 1h — Interface in active state 2h — Interface in PARTIAL power management state 6h — Interface in SLUMBER power management state All other values reserved.			RO
07 :04	SPD	Current Interface Speed — Indicates the negotiated interface communication speed. **Value** / **Description** 0h — Device not present or communication not established 1h — Generation 1 communication rate negotiated 2h — Generation 2 communication rate negotiated All other values reserved. The PCH supports Gen 1 communication rates (1.5 Gb/s) and Gen 2 rates (3.0 Gb/s).			RO

Table 8-108. Offset 328h: Port [4:5] Serial ATA Status Register (B0:D31:F2) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 328h + 80h **Offset End:** 32Bh + 80h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
03 :00	DET	Device Detection — Indicates the interface device detection and Phy state: **Value** / **Description** 0h — No device detected and Phy communication not established 1h — Device presence detected but Phy communication not established 3h — Device presence detected and Phy communication established 4h — Phy in offline mode as a result of the interface being disabled or running in a BIST loopback mode All other values reserved.			RO

8.4.2.11 Offset 32Ch: Port [4:5] Serial ATA Control Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 32Ch
Port 5: ABAR + 3ACh

This is a 32-bit read-write register by which software controls SATA* capabilities. Writes to the SControl register result in an action being taken by the PCH or the interface. Reads from the register return the last value written to it.

Table 8-109. Offset 32Ch: Port [4:5] Serial ATA Control Register (B0:D31:F2) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** ABAR	**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 32Ch + 80h **Offset End:** 32Fh + 80h	
Size: 32 bit	**Default:** 000000004h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved			
19 :16	Reserved	Reserved			
15 :12	SPM	Select Power Management — This field is not used by AHCI			RW
11 :08	IPM	Interface Power Management Transitions Allowed — Indicates which power states the PCH is allowed to transition to: **Value** / **Description** 0h — No interface restrictions 1h — Transitions to the PARTIAL state disabled 2h — Transitions to the SLUMBER state disabled 3h — Transitions to both PARTIAL and SLUMBER states disabled All other values reserved			RW
07 :04	SPD	Speed Allowed — Indicates the highest allowable speed of the interface. This speed is limited by the CAP.ISS (ABAR+00h:bit 23:20) field. **Value** / **Description** 0h — No speed negotiation restrictions 1h — Limit speed negotiation to Generation 1 communication rate 2h — Limit speed negotiation to Generation 2 communication rate The PCH Supports Gen 1 communication rates (1.5 Gb/s) and Gen 2 rates (3.0 Gb/s).			RW

Table 8-109. Offset 32Ch: Port [4:5] Serial ATA Control Register (B0:D31:F2) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 32Ch + 80h **Offset End:** 32Fh + 80h	
Size: 32 bit	**Default:** 000000004h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
03 :00	DET	Device Detection Initialization — Controls the PCH's device detection and interface initialization. **Value** / **Description** 0h — No device detection or initialization action requested 1h — Perform interface communication initialization sequence to establish communication. This is functionally equivalent to a hard reset and results in the interface being reset and communications re-initialized 4h — Disable the Serial ATA interface and put Phy in offline mode All other values reserved. When this field is written to a 1h, the PCH initiates COMRESET and starts the initialization process. When the initialization is complete, this field shall remain 1h until set to another value by software. This field may only be changed to 1h or 4h when PxCMD.ST is 0. Changing this field while the PCH is running results in undefined behavior. Note: It is permissible to implement any of the Serial ATA defined behaviors for transmission of COMRESET when DET=1h.			RW

8.4.2.12 Offset 330h: Port [4:5] Serial ATA Error Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 330h
Port 5: ABAR + 3B0h

Bits 26:16 of this register contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes. Bits 11:0 contain error information used by host software in determining the appropriate response to the error condition. If one or more of bits 11:8 of this register are set, the controller will stop the current transfer.

Table 8-110. Offset 330h: Port [4:5] Serial ATA Error Register (B0:D31:F2) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 330h + 80h **Offset End:** 333h + 80h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:27	Reserved	Reserved				
26	X	Exchanged — When set to one this bit indicates that a change in device presence has been detected since the last time this bit was cleared. This bit shall always be set to 1 anytime a COMINIT signal is received. This bit is reflected in the P0IS.PCS bit.				RWC
25	F	Unrecognized FIS Type — Indicates that one or more FISs were received by the Transport layer with good CRC, but had a type field that was not recognized.				RWC
24	T	Transport state transition error — Indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared.				RWC
23	T	Transport state transition error — Indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared.				RWC
22	H	Handshake — Indicates that one or more R_ERR handshake response was received in response to frame transmission. Such errors may be the result of a CRC error detected by the recipient, a disparity or 8b/10b decoding error, or other error condition leading to a negative handshake on a transmitted frame.				RWC
21	C	CRC Error — Indicates that one or more CRC errors occurred with the Link Layer.				RWC
20	D	Disparity Error — This field is not used by AHCI.				RWC
19	B	10b to 8b Decode Error — Indicates that one or more 10b to 8b decoding errors occurred.				RWC
18	W	Comm Wake — Indicates that a Comm Wake signal was detected by the Phy.				RWC
17	I	Phy Internal Error — Indicates that the Phy detected some internal error.				RWC
16	N	PhyRdy Change — When set to 1 this bit indicates that the internal PhyRdy signal changed state since the last time this bit was cleared. In the PCH, this bit will be set when PhyRdy changes from a 0 -> 1 or a 1 -> 0. The state of this bit is then reflected in the PxIS.PRCS interrupt status bit and an interrupt will be generated if enabled. Software clears this bit by writing a 1 to it.				RWC
15:12	Reserved	Reserved				

Table 8-110. Offset 330h: Port [4:5] Serial ATA Error Register (B0:D31:F2) (Sheet 2 of 2)

Description:							
View: PCI		**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 330h + 80h **Offset End:** 333h + 80h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
	11	E	Internal Error — The SATA* controller failed due to a master or target abort when attempting to access system memory.				RWC
	10		Protocol Error — A violation of the Serial ATA protocol was detected. ***Note:*** The PCH does not set this bit for all protocol violations that may occur on the SATA* link.				RWC
	09	C	Persistent Communication or Data Integrity Error — A communication error that was not recovered occurred that is expected to be persistent. Persistent communications errors may arise from faulty interconnect with the device, from a device that has been removed or has failed, or a number of other causes.				RWC
	08	T	Transient Data Integrity Error — A data integrity error occurred that was not recovered by the interface.				RWC
07	:02	Reserved	Reserved				
	01	M	Recovered Communications Error — Communications between the device and host was temporarily lost but was re-established. This can arise from a device temporarily being removed, from a temporary loss of Phy synchronization, or from other causes and may be derived from the PhyNRdy signal between the Phy and Link layers.				RWC
	00	I	Recovered Data Integrity Error — A data integrity error occurred that was recovered by the interface through a retry operation or other recovery action.				RWC

8.4.2.13 Offset 334h: Port [4:5] Serial ATA Active (B0:D31:F2)

Address Offset: Port 4: ABAR + 334h
Port 5: ABAR + 3B4h

Table 8-111. Offset 334h: Port [4:5] Serial ATA Active (B0:D31:F2)

Description:						
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2		**Offset Start:** 334h + 80h **Offset End:** 337h + 80h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DS	Device Status — System software sets this bit for SATA* queuing operations prior to setting the PxCI.CI bit in the same command slot entry. This field is cleared via the Set Device Bits FIS. This field is also cleared when PxCMD.ST (ABAR+118h/198h/218h/298h:bit 0) is cleared by software, and as a result of a COMRESET or SRST.				RW

8.4.2.14 Offset 338h: Port [4:5] Command Issue Register (B0:D31:F2)

Address Offset: Port 4: ABAR + 338h
Port 5: ABAR + 3B8h

Table 8-112. Offset 338h: Port [4:5] Command Issue Register (B0:D31:F2)

Description:					
View: PCI	**BAR:** ABAR		**Bus:Device:Function:** B0:D31:F2	**Offset Start:** 338h + 80h **Offset End:** 33Bh + 80h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CI	Commands Issued — This field is set by software to indicate to the PCH that a command has been built-in system memory for a command slot and may be sent to the device. When the PCH receives a FIS which clears the BSY and DRQ bits for the command, it clears the corresponding bit in this register for that command slot. Bits in this field shall only be set to '1' by software when PxCMD.ST is set to '1'. This field is also cleared when PxCMD.ST (ABAR+118h/198h/218h/298h:bit 0) is cleared by software.			RW

§ §

9.0 SATA* Controller Registers (B0:D31:F5)

9.1 PCI Configuration Registers (SATA–B0:D31:F5)

Note: Address locations that are not shown should be treated as Reserved.

All of the SATA* registers are in the core well. None of the registers can be locked.

9.1.1 SATA* Controller PCI Register Address Map

Table 9-1. SATA* Controller PCI Register Address Map (Sheet 1 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: VID: Vendor Identification Register (SATA—B0:D31:F5)" on page 507	8086h
02h	03h	"Offset 02h: Device Identification Register (SATA—B0:D31:F5)" on page 508	2326h/23A6h [1]
04h	05h	"Offset 04h: PCI Command Register (SATA—B0:D31:F5)" on page 508	0007h
06h	07h	"Offset 06h: PCI Status Register (SATA—B0:D31:F5)" on page 509	02B0h
08h	08h	"Offset 08h: RID—Revision Identification Register (SATA—B0:D31:F5)" on page 510	See register description
09h	09h	"Offset 09h: Programming Interface Register (SATA—B0:D31:F5)" on page 511	00h
0Ah	0Ah	"Offset 0Ah: Sub Class Code Register (SATA—B0:D31:F5)" on page 511	01h
0Bh	0Bh	"Offset 0Bh: BCC—Base Class Code Register (SATA–B0:D31:F5)" on page 512	01h
0Dh	0Dh	"Offset 0Dh: Primary Master Latency Timer Register (SATA–B0:D31:F5)" on page 512	00h
10h	13h	"Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F5)" on page 513	00000001h
14h	17h	"Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F5)" on page 514	00000001h
18h	1Bh	"Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1)" on page 514	00000001h
1Ch	1Fh	"Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F5)" on page 515	00000001h
20h	23h	"Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F5)" on page 515	00000001h
24h	27h	"Offset 24h: SATA Index/Data Pair Base Address Register (SATA–B0:D31:F5)" on page 516	00000000h
2Ch	2Dh	"Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F5)" on page 516	0000h
2Eh	2Fh	"Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F5)" on page 517	0000h
34h	34h	"Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F5)" on page 517	70h
3Ch	3Ch	"Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F5)" on page 517	00h

Table 9-1. SATA* Controller PCI Register Address Map (Sheet 2 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
3Dh	3Dh	"Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F5)" on page 518	See register description
40h	43h	"Offset 40h: IDE Timing Register (SATA–B0:D31:F5)" on page 518	0000h
48h	48h	"Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F5)" on page 519	00h
4Ah	4Bh	"Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F5)" on page 519	0000h
54h	57h	"Offset 54h: IDE I/O Configuration Register (SATA–B0:D31:F5)" on page 520	00000000h
70h	71h	"Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F5)" on page 521	B001h
72h	73h	"Offset 72h: PC—PCI Power Management Capabilities Register (SATA–B0:D31:F5)" on page 521	4003h
74h	75h	"Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F5)" on page 522	0008h
90h	90h	"Offset 90h: MAP—Address Map Register (SATA–B0:D31:F5)" on page 523	00h
92h	93h	"Offset 92h: Port Control and Status Register (SATA–B0:D31:F5)" on page 524	0000h
A8h	ABh	"Offset A8h: SATA Capability Register 0 (SATA–B0:D31:F5)" on page 525	0010B012h
ACh	AFh	"Offset ACh: SATA* Capability Register 1 (SATA–B0:D31:F5)" on page 525	00000048h
B0h	B1h	"Offset B0h: FLR Capability ID (SATA–B0:D31:F5)" on page 526	0009h
B2h	B3h	"Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5)" on page 526	2006h
B2h	B3h	"Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5)" on page 527	2006h
B4h	B5h	"Offset B4h: FLR Control (SATA–B0:D31:F5)" on page 527	0000h
C0h	C0h	"Offset C0h: APM Trapping Control Register (SATA–B0:D31:F5)" on page 528	00h
C4h	C4h	"Offset C4h: APM Trapping Control Register (SATA–B0:D31:F5)" on page 528	00h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

Note: The PCH SATA* controller is not arbitrated as a PCI device, therefore it does not need a master latency timer.

9.1.1.1 Offset 00h: Vendor Identification Register (SATA–B0:D31:F5)

Table 9-2. Offset 00h: VID: Vendor Identification Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

9.1.1.2 Offset 02h: Device Identification Register (SATA–B0:D31:F5)

Table 9-3. Offset 02h: Device Identification Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2326h/23A6h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device ID — This is a 16-bit value assigned to the PCH SATA* controller.		2326h/23A6h[1]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

9.1.1.3 Offset 04h: PCI Command Register (SATA–B0:D31:F5)

Table 9-4. Offset 04h: PCI Command Register (SATA—B0:D31:F5) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0007h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		00h	
10	INTx_Disable	Interrupt Disable — This disables pin-based INTx# interrupts. This bit has no effect on MSI operation. 0 = Internal INTx# messages are generated if there is an interrupt and MSI is not enabled. 1 = Internal INTx# messages will not be generated.		0h	RW
09	FBE	Fast Back to Back Enable — Reserved as 0.		0h	RO
08	SERR_EN	SERR# Enable — Reserved as 0.		0h	RO
07	WCC	Wait Cycle Control — Reserved as 0.		0h	RO
06	PER	Parity Error Response: 0 = Disabled. SATA* controller will not generate PERR# when a data parity error is detected. 1 = Enabled. SATA* controller will generate PERR# when a data parity error is detected.		0h	RW
05	VPS	VGA Palette Snoop — Reserved as 0.		0h	RO
04	PMWE	Postable Memory Write Enable — Reserved as 0.		0h	RO
03	SCE	Special Cycle Enable — Reserved as 0.		0h	RO

Table 9-4. Offset 04h: PCI Command Register (SATA—B0:D31:F5) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0007h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BME	Bus Master Enable — This bit controls the PCH's ability to act as a PCI master for IDE Bus Master transfers. This bit does not impact the generation of completions for split transaction commands.		1	RW
01	MSE	Memory Space Enable — This controller does not support AHCI, therefore no memory space is required.		1	RO
00	IOSE	I/O Space Enable — This bit controls access to the I/O space registers. 0 = Disables access to the Legacy or Native IDE ports (both Primary and Secondary) as well as the Bus Master I/O registers. 1 = Enable. The Base Address register for the Bus Master registers should be programmed before this bit is set.		1	RW

9.1.1.4 Offset 06h: PCI Status Register (SATA–B0:D31:F5)

Table 9-5. Offset 06h: PCI Status Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 02B0h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the Parity Error Response bit (B0:D31:F5: Offset 04h bit 6) is not set.		0h	RWC
14	SSE	Signaled System Error — Set when the LPC bridge signals a system error to the internal SERR# logic.		0h	RWC
13	RMA	Received Master Abort: 0 = No master abort received. 1 = Set when the bridge receives a master abort status from the I/O data bus.		0h	RWC
12	Reserved	Reserved			
11	STA	Signaled Target Abort — Reserved as 0.		0h	RO
10 :09	DEV_STS	DEVSEL# Timing Status: 01 = Hardwired; Controls the device select time for the SATA* controller's PCI interface.		00h	RO
08	DPED	Data Parity Error Detected — For PCH, this bit can only be set on read completions received from SiBUS where there is a parity error. 1 = SATA* controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set.		0h	RWC

Table 9-5. Offset 06h: PCI Status Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 02B0h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	FB2BC	Fast Back to Back Capable — Reserved as 1.		0h	RO
06	UDF	User Definable Features — Reserved as 0.		0h	RO
05	66MHZ_CAP	66MHz Capable — Reserved as 1.		0h	RO
04	CAP_LIST	Capabilities List — This bit indicates the presence of a capabilities list. The minimum requirement for the capabilities list must be PCI power management for the SATA* controller.		1	RO
03	INTS	Interrupt Status — Reflects the state of INTx# messages, IRQ14 or IRQ15. 0 = Interrupt is cleared (independent of the state of Interrupt Disable bit in the command register [offset 04h]). 1 = Interrupt is to be asserted		0h	RO
2 :0	Reserved	Reserved			

9.1.1.5 Offset 08h: RID—Revision Identification Register (SATA–B0:D31:F5)

Table 9-6. Offset 08h: RID—Revision Identification Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See bit description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: This is an 8-bit value indicating the stepping of the SATA* controller hardware		Variable	RO

9.1.1.6 Offset 09h: Programming Interface Register (SATA–B0:D31:F5)

Table 9-7. Offset 09h: Programming Interface Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		This read-only bit is a 1 to indicate that the PCH supports bus master operation			RO
06 :04	Reserved	Reserved			
03	SNC	Secondary Mode Native Capable — Indicates whether or not the secondary channel has a fixed mode of operation. 0 = Indicates the mode is fixed and is determined by the (read-only) value of bit 2. This bit will always return '0'.			RO
02	SNE	Secondary Mode Native Enable: Determines the mode that the secondary channel is operating in. 1 = Secondary controller operating in native PCI mode. This bit will always return '1'.			RO
01	PNC	Primary Mode Native Capable — Indicates whether or not the primary channel has a fixed mode of operation. 0 = Indicates the mode is fixed and is determined by the (read-only) value of bit 0. This bit will always return '0'.			RO
00	PNE	Primary Mode Native Enable: Determines the mode that the primary channel is operating in. 1 = Primary controller operating in native PCI mode. This bit will always return '1'.			RO

9.1.1.7 Offset 0Ah: Sub Class Code Register (SATA–B0:D31:F5)

Table 9-8. Offset 0Ah: Sub Class Code Register (SATA—B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code: The value of this field determines whether the controller supports legacy IDE mode.			RO

9.1.1.8 Offset 0Bh: BCC—Base Class Code Register (SATA–B0:D31:F5)

Table 9-9. Offset 0Bh: BCC—Base Class Code Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code: 01h = Mass storage device.			RO

9.1.1.9 Offset 0Dh: Primary Master Latency Timer Register (SATA–B0:D31:F5)

Table 9-10. Offset 0Dh: Primary Master Latency Timer Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MLTC	Master Latency Timer Count: 00h = Hardwired. The SATA* controller is implemented internally, and is not arbitrated as a PCI device, so it does not need a Master Latency Timer.			RO

9.1.1.10 Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F5)

This 8-byte I/O space is used in native mode for the Primary Controller's Command Block.

Table 9-11. Offset 10h: Primary Command Block Base Address Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :03	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
02 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.11 Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F5)

This 4-byte I/O space is used in native mode for the Primary Controller's Command Block.

Table 9-12. Offset 14h: Primary Control Block Base Address Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :03	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
02 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.12 Offset 18h: SCMD-Secondary Command Block Base Address Register (SATA–B0:D31:F5)

This 8-byte I/O space is used in native mode for the Secondary Controller's Command Block.

Table 9-13. Offset 18h: Secondary Command Block Base Address Register (IDE D31:F1)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :03	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
02 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.13 Offset 1Ch: SCNL- Secondary Control Block Base Address Register (SATA–B0:D31:F5)

This 4-byte I/O space is used in native mode for the Secondary Controller's Command Block.

Table 9-14. Offset 1Ch: Secondary Control Block Base Address Register (IDE B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :02	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.14 Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F5)

The Bus Master IDE interface function uses Base Address register 5 to request a 16-byte IO space to provide a software interface to the Bus Master functions. Only 12 bytes are actually used (6 bytes for primary, 6 bytes for secondary). Only bits [15:04] are used to decode the address.

Table 9-15. Offset 20h: Legacy Bus Master Base Address Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			O
15 :05	BA	Base Address — This field provides the base address of the I/O space (8 consecutive I/O locations).			RW
04	BA4	Base Address 4 — When SCC is 01h, this bit will be R/W resulting in requesting 16B of I/O space.			RW
03 :01	Reserved	Reserved			O
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.15 Offset 24h: SATA Index/Data Pair Base Address Register (SATA–B0:D31:F5)

Table 9-16. Offset 24h: SATA Index/Data Pair Base Address Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :04	BA	Base Address — Base address of register I/O space			RW
03 :01	Reserved	Reserved			
00	RTE	Resource Type Indicator — Hardwired to 1 to indicate a request for I/O space.			RO

9.1.1.16 Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F5)

Table 9-17. Offset 2Ch: Subsystem Vendor Identification Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	Subsystem Vendor ID — Value is written by BIOS. No hardware action taken on this value.			RWO

9.1.1.17 Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F5)

Table 9-18. Offset 2Eh: Subsystem Identification Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	Subsystem ID — Value is written by BIOS. No hardware action taken on this value.			RWO

9.1.1.18 Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F5)

Table 9-19. Offset 34h: Capabilities Pointer Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 70h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CAP_PTR	Capabilities Pointer — Indicates that the first capability pointer offset is 70h if the Sub Class Code (SCC) (Dev 31:F2:0Ah) is configure as IDE mode (value of 01).			RO

9.1.1.19 Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F5)

Table 9-20. Offset 3Ch: Interrupt Line Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IL	Interrupt Line — This field is used to communicate to software the interrupt line that the interrupt pin is connected to. These bits are not reset by FLR.			RW

9.1.1.20 Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F5)

Table 9-21. Offset 3Dh: Interrupt Pin Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IP	Interrupt Pin — This reflects the value of D31IP.SIP1 (Chipset Config Registers:Offset 3100h:bits 11:8).			RO

9.1.1.21 Offset 40h: IDE Timing Register (SATA–B0:D31:F5)

Address Offset: Primary: 40h–41h

Secondary: 42h–43h

Table 9-22. Offset 40h: IDE Timing Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 40h **Offset End:** 43h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	IDE	IDE Decode Enable — Individually enable/disable the Primary or Secondary decode. 0 = Disable. 1 = Enables the PCH to decode the associated Command Blocks (1F0–1F7h for primary, 170–177h for secondary) and Control Block (3F6h for primary and 376h for secondary). This bit effects the IDE decode ranges for both legacy and native-Mode decoding. This bit affects SATA* operation in both combined and non-combined ATA modes.			RW
14 :00	Reserved	Reserved			

9.1.1.22 Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F5)

Address Offset: Primary: 48h

Table 9-23. Offset 48h: Synchronous DMA Control Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 48h **Offset End:** 48h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	IReserved			
03 :00	SDMA_CNT	**SDMA_CNT Field 1**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

9.1.1.23 Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F5)

Address Offset: Primary: 4Ah–4Bh

Table 9-24. Offset 4Ah: Synchronous DMA Timing Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 4Ah **Offset End:** 4Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09 :08	SDMA_TIM2	**SDMA_TIM Field 2**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
07 :02	Reserved	Reserved			
01 :00	SDMA_TIM1	**SDMA_TIM Field 1—**R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

9.1.1.24 Offset 54h: IDE I/O Configuration Register (SATA–B0:D31:F5)

Address Offset: Primary: 54h–57h

Table 9-25. Offset 4Ah: IDE I/O Configuration Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :16	IDE_CFG6	**IDE_CONFIG Field 6**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
15	Reserved	Reserved			
14	IDE_CFG5	**IDE_CONFIG Field 5**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
13	Reserved	Reserved			
12	IDE_CFG4	**IDE_CONFIG Field 4**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
11 :08	Reserved	Reserved			
07 :04	IDE_CFG3	**IDE_CONFIG Field 3**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
03	Reserved	Reserved			
02	IDE_CFG2	**IDE_CONFIG Field 2**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW
01	Reserved	Reserved			
00	IDE_CFG1	**IDE_CONFIG Field 1**—R/W. This field is R/W to maintain software compatibility. This field has no effect on hardware.			RW

9.1.1.25 Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F5)

Table 9-26. Offset 70h: PCI Power Management Capability Identification Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 70h **Offset End:** 71h	
Size: 16 bit	**Default:** B001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — When SCC is 01h, this field will be B0h indicating the next item is FLR Capability Pointer in the list.			RO
07 :00	CID	Capability ID — Indicates that this pointer is a PCI power management.			RO

9.1.1.26 Offset 72h: PC—PCI Power Management Capabilities Register (SATA–B0:D31:F5)

Table 9-27. Offset 72h: PC—PCI Power Management Capabilities Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 72h **Offset End:** 73h	
Size: 16 bit	**Default:** 4003h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	PME_SUP	PME Support — By default with SCC = 01h, the default value of 00000 indicates no PME support in IDE mode.			RO
10	D2_SUP	D2 Support — Hardwired to 0. The D2 state is not supported			RO
09	D1_SUP	D1 Support — Hardwired to 0. The D1 state is not supported			RO
08 :06	AUX_CUR	Auxiliary Current — PME# from $D3_{COLD}$ state is not supported, therefore this field is 000b.			RO
05	DSI	Device Specific Initialization — Hardwired to 0 to indicate that no device-specific initialization is required.			RO
04	Reserved	Reserved			
03	PME_CLK	PME Clock — Hardwired to 0 to indicate that PCI clock is not required to generate PME#.			RO
02 :00	VER	Version — Hardwired to 011 to indicates support for Revision 1.2 of the PCI Power Management Specification.			RO

9.1.1.27 Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F5)

Table 9-28. Offset 74h: PCI Power Management Control and Status Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 74h **Offset End:** 75h	
Size: 16 bit	**Default:** 0008h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	PMES	PME Status — Bit is set when a PME event is to be requested, and if this bit and PMEE is set, a PME# will be generated from the SATA* controller. Note: When SCC=01h this bit will be RO '0'. Software is advised to clear PMEE together with PMES prior to changing SCC through MAP.SMS. This bit is not reset by Function Level Reset.			RWC
14 :09	Reserved	Reserved			
08	PMEE	PME Enable — When SCC is not 01h, this bit R/W. When set, the SATA* controller generates PME# form $D3_{HOT}$ on a wake event. Note: When SCC=01h this bit will be RO '0'. Software is advised to clear PMEE together with PMES prior to changing SCC through MAP.SMS. This bit is not reset by Function Level Reset.			RW
07 :04	Reserved	Reserved			
03	NSFRST	No Soft Reset — These bits are used to indicate whether devices transitioning from $D3_{HOT}$ state to D0 state will perform an internal reset. 0 = Device transitioning from $D3_{HOT}$ state to D0 state perform an internal reset. 1 = Device transitioning from $D3_{HOT}$ state to D0 state do not perform an internal reset. Configuration content is preserved. Upon transition from the $D3_{HOT}$ state to D0 state initialized state, no additional operating system intervention is required to preserve configuration context beyond writing to the PowerState bits. Regardless of this bit, the controller transition from $D3_{HOT}$ state to D0 state by a system or bus segment reset will return to the state D0 uninitialized with only PME context preserved if PME is supported and enabled.			RO
02	Reserved	Reserved			
01 :00	PS	Power State — These bits are used both to determine the current power state of the SATA* controller and to set a new power state. 00 = D0 state 11 = $D3_{HOT}$ state When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked.			RW

9.1.1.28 Offset 90h: MAP—Address Map Register (SATA–B0:D31:F5)

Table 9-29. Offset 90h: MAP—Address Map Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 90h **Offset End:** 90h	
Size: 16 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09 :08	SPD	SATA* Port Disable — SW programs these bits to disable a SATA* port on the controller. 1 = Corresponding port is disabled. [Bit 8 = Port 4 and Bit 9 = Port 5] 0 = Corresponding port is enabled. Note: To ensure a port is properly disabled, BIOS shall configure MAP.SPD first and then configure PCS.PxE. In order to securely prevent port 4 and/or port 5 from being enabled, BIOS shall ensure the following: For Port 4 - B0:D31:F2 MAP[12]= 1 and D31F5 MAP[8]= 1 For Port 5 - B0:D31:F2 MAP[13]= 1 and D31F5 MAP[9]= 1 This field is nor reset by Function Level Reset.			RWO
07 :06	SMS	SATA* Mode Select — Software programs these bits to control the mode in which the SATA* Controller should operate. 00b = IDE Mode All other combinations are reserved.			RW
05 :02	Reserved	Reserved			
01 :00	MV	Map Value — Reserved.			

9.1.1.29 Offset 92h: Port Control and Status Register (SATA–B0:D31:F5)

By default, the SATA* ports are set to the disabled state (bits [5:0] = '0'). When enabled by software, the ports can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the "off" state and cannot detect any devices.

If an AHCI-aware enabled operating system is being booted then system BIOS shall insure that all supported SATA* ports are enabled prior to passing control to the OS. Once the AHCI aware OS is booted it becomes the enabling/disabling policy owner for the individual SATA* ports. This is accomplished by manipulating a port's PxSCTL and PxCMD fields. Because an AHCI aware OS will typically not have knowledge of the PxE bits and because the PxE bits act as master on/off switches for the ports, pre-boot software must insure that these bits are set to '1' prior to booting the OS, regardless as to whether or not a device is currently on the port.

Table 9-30. Offset 92h: Port Control and Status Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 92h **Offset End:** 93h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09	P5P	Port 5 Present — The status of this bit may change at any time. This bit is cleared when the port is disabled via P1E. This bit is not cleared upon surprise removal of a device. 0 = No device detected. 1 = The presence of a device on Port 1 has been detected.			RO
08	P4P	Port 4 Present — The status of this bit may change at any time. This bit is cleared when the port is disabled via P0E. This bit is not cleared upon surprise removal of a device. 0 = No device detected. 1 = The presence of a device on Port 0 has been detected.			RO
07 :02	Reserved	Reserved			
01	P5E	Port 5 Enabled: 0 = Disabled. The port is in the 'off' state and cannot detect any devices. 1 = Enabled. The port can transition between the on, partial, and slumber states and can detect devices. This bit is read-only '0' when MAP.SPD[1]= 1.			RW
00	P4E	Port 4 Enabled: 0 = Disabled. The port is in the 'off' state and cannot detect any devices. 1 = Enabled. The port can transition between the on, partial, and slumber states and can detect devices. This bit is read-only '0' when MAP.SPD[0]= 1.			RW

9.1.1.30 Offset A8h: SATA Capability Register 0 (SATA–B0:D31:F5)

Table 9-31. Offset A8h: SATA Capability Register 0 (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** A8h **Offset End:** ABh	
Size: 32 bit	**Default:** 0010B012h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :20	MAJREV	Major Revision — Major revision number of the SATA* Capability Pointer implemented.			RO
19 :16	MINREV	Minor Revision — Minor revision number of the SATA* Capability Pointer implemented.			RO
15 :08	NEXT	Next Capability Pointer — Points to the next capability structure.			RWO
07 :00	CAP	Capability ID — The value of 12h has been assigned by the PCI SIG to designate the SATA* capability pointer.			RO

9.1.1.31 Offset ACh: SATA* Capability Register 1 (SATA–B0:D31:F5)

Table 9-32. Offset ACh: SATA* Capability Register 1 (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** ACh **Offset End:** AFh	
Size: 32 bit	**Default:** 00000048h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :04	BAROFST	BAR Offset — Indicates the offset into the BAR where the index/Data pair are located (in DWord granularity). The index and Data I/O registers are located at offset 10h within the I/O space defined by LBAR (BAR4). A value of 004h indicates offset 10h.			RO
03 :00	BARLOC	BAR Location — Indicates the absolute PCI Configuration Register address of the BAR containing the Index/Data pair (in DWord granularity). The Index and Data I/O registers reside within the space defined by LBAR (BAR4) in the SATA* controller. a value of 8h indicates and offset of 20h, which is LBAR (BAR4).			RO

9.1.1.32 Offset B0h: FLR Capability ID (SATA–B0:D31:F5)

Table 9-33. Offset B0h: FLR Capability ID (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** B0h **Offset End:** B1h	
Size: 16 bit	**Default:** 0009h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08		Next Capability Pointer — A value of 00h indicates the final item in the Capability List.			RO
07 :00		Capability ID — The value of this field depends on the FLRCSSECL bit. If FLRCSSEL = 0, this field is 13h If FLRCSSEL = 1, this field is 09h, indicating vendor specific capability.			RO

9.1.1.33 Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5)

Address Offset: B2h-B3h
Attribute: RO, RWO
Default Value: 2006h
Size: 16 bits
Function Level Reset: No (Bits 9:8 only)

When FLRCSSEL = '0', this register is defined as follows:

Table 9-34. Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 2006h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09		FLR Capability — This field indicates support for Function Level Reset.			RWO
08		TXP Capability — This field indicates support for the Transactions Pending (TXP) bit. TXP must be supported if FLR is supported.			RWO
07 :00		Capability Length — This field indicates the number of bytes of the Vendor Specific capability as required by the PCI spec. It has the value of 06h for FLR Capability.			RO

When FLRCSSEL = '1', this register is defined as follows:

Table 9-35. Offset B2h: FLR Capability Length and Value (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 2006h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12		Vendor Specific Capability ID — A value of 02h identifies this capability as a Function Level Reset.			RO
11 :08		Capability Version — This field indicates the version of the FLR capability.			RO
07 :00		Capability Length — This field indicates the number of bytes of the Vendor Specific capability as required by the PCI spec. It has the value of 06h for FLR Capability.			RO

9.1.1.34 Offset B4h: FLR Control (SATA–B0:D31:F5)

Table 9-36. Offset B4h: FLR Control (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** B4h **Offset End:** B5h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved			
08	TXP	Transactions Pending: 0 = Completions for all Non-Posted requests have been received by the controller. 1 = Controller has issued Non-Posted request which has not been completed.			RO
07 :01	Reserved	Reserved			
00		Initiate FLR — Used to initiate FLR transition. A write of '1' indicates FLR transition.			RW

9.1.1.35 Offset C0h: APM Trapping Control Register (SATA–B0:D31:F5)

This SATA* controller does not support legacy I/O access. Therefore, this register is reserved. Software shall not change the default values of the register; otherwise the result will be undefined.

Table 9-37. Offset C0h: APM Trapping Control Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** C0h **Offset End:** C0h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Reserved	Reserved			

9.1.1.36 Offset C4h: APM Trapping Control Register (SATA–B0:D31:F5)

This SATA* controller does not support legacy I/O access. Therefore, this register is reserved. Software shall not change the default values of the register; otherwise the result will be undefined.

Table 9-38. Offset C4h: APM Trapping Control Register (SATA–B0:D31:F5)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** C4h **Offset End:** C4h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Reserved	Reserved			

9.1.2 Bus Master IDE I/O Registers (B0:D31:F5)

The bus master IDE function uses 16 bytes of I/O space, allocated via the LBAR register, located in Bus 0:Device 31:Function 5 Configuration space, offset 20h. All bus master IDE I/O space registers can be accessed as byte, word, or dword quantities. Reading reserved bits returns an indeterminate, inconsistent value, and writes to reserved bits have no affect (but should not be attempted). These registers are only used for legacy (IDE) operation. Software must not use these registers when running AHCI. The description of the I/O registers is shown in the following table.

Table 9-39. Bus Master IDE I/O Registers (B0:D31:F5)

Offset Start	Offset End	Register ID - Description	Default Value
00h	08h	"Offset 00h: Bus Master IDE Command Register (B0:D31:F5)" on page 529	00h
02h	0Ah	"Offset 02h: Bus Master IDE Status Register (B0:D31:F5)" on page 531	00h
0Ch	0Fh	"Offset 04h: Bus Master IDE Descriptor Table Pointer Register (B0:D31:F5)" on page 532	00h

9.1.2.1 Offset 00h: Bus Master IDE Command Register (B0:D31:F5)

Address Offset:

Primary: BAR + 00hSecondary: BAR + 08h

Table 9-40. Offset 00h: Bus Master IDE Command Register (B0:D31:F5)

Description:					
View: PCI	**BAR:** LBAR (IO)		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 00h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Reserved	Reserved			

Table 9-40. Offset 00h: Bus Master IDE Command Register (B0:D31:F5)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 00h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:03	R/WC	Read / Write Control — This bit sets the direction of the bus master transfer: This bit must NOT be changed when the bus master function is active. 0 = Memory reads 1 = Memory writes			RW
02 :01	Reserved	Reserved			
:00	START	Start/Stop Bus Master: 0 = All state information is lost when this bit is cleared. Master mode operation cannot be stopped and then resumed. If this bit is reset while bus master operation is still active (for example, the Bus Master IDE Active bit (B0:D31:F5:BAR + 02h, bit 0) of the Bus Master IDE Status register for that IDE channel is set) and the drive has not yet finished its data transfer (the Interrupt bit in the Bus Master IDE Status register for that IDE channel is not set), the bus master command is said to be aborted and data transferred from the drive may be discarded instead of being written to system memory. 1 = Enables bus master operation of the controller. Bus master operation does not actually start unless the Bus Master Enable bit (D31:F1:04h, bit 2) in PCI configuration space is also set. Bus master operation begins when this bit is detected changing from 0 to 1. The controller will transfer data between the IDE device and memory only when this bit is set. Master operation can be halted by writing a 0 to this bit. This bit is intended to be cleared by software after the data transfer is completed, as indicated by either the Bus Master IDE Active bit being cleared or the Interrupt bit of the Bus Master IDE Status register for that IDE channel being set, or both. Hardware does not clear this bit automatically. If this bit is cleared to 0 prior to the DMA data transfer being initiated by the drive in a device to memory data transfer, then the PCH will not send DMAT to terminate the data transfer. SW intervention (for example, sending SRST) is required to reset the interface in this condition.			RW

9.1.2.2 Offset 02h: Bus Master IDE Status Register (B0:D31:F5)

Address Offset:

Primary: BAR + 02hSecondary: BAR + 0Ah

Table 9-41. Offset 02h: Bus Master IDE Status Register (B0:D31:F5)

Description:					
View: PCI	**BAR:** LBAR (IO)	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 02h **Offset End:** 0Ah	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	PRDIS	PRD Interrupt Status: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the host controller execution of a PRD that has its PRD_INT bit set.			RWC
06	Reserved	Reserved			
05		Drive 0 DMA Capable: 0 = Not Capable 1 = Capable. Set by device dependent code (BIOS or device driver) to indicate that drive 0 for this channel is capable of DMA transfers, and that the controller has been initialized for optimum performance. The PCH does not use this bit. It is intended for systems that do not attach BMIDE to the PCI bus.			RW
04 :03	Reserved	Reserved			
02		Interrupt: 0 = Software clears this bit by writing a 1 to it. 1 = Set when a device FIS is received with the 'I' bit set, provided that software has not disabled interrupts via the IEN bit of the Device Control Register (see chapter 5 of the *Serial ATA Specification*, Revision 1.0a).			RWC
01		Error: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set when the controller encounters a target abort or master abort when transferring data on PCI.			RWC
00	ACT	Bus Master IDE Active: 0 = This bit is cleared by the PCH when the last transfer for a region is performed, where EOT for that region is set in the region descriptor. It is also cleared by the PCH when the Start Bus Master bit (B0:D31:F5:BAR+ 00h, bit 0) is cleared in the Command register. When this bit is read as a 0, all data transferred from the drive during the previous bus master command is visible in system memory, unless the bus master command was aborted. 1 = Set by the PCH when the Start bit is written to the Command register.			RO

9.1.2.3 Offset 04h: Bus Master IDE Descriptor Table Pointer Register (B0:D31:F5)

Address Offset: Primary: BAR + 04h–07hSecondary: BAR + 0Ch–0Fh

Table 9-42. Offset 04h: Bus Master IDE Descriptor Table Pointer Register (B0:D31:F5)

Description:					
View: PCI	**BAR:** LBAR (IO)		**Bus:Device:Function:** B0:D31:F5	**Primary Offset Start:** 04h **Offset End:** 07h **Secondary Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address of Descriptor Table — The bits in this field correspond to bits [31:2] of the memory location of the Physical Region Descriptor (PRD). The Descriptor Table must be dword-aligned. The Descriptor Table must not cross a 64-K boundary in memory.			RW
01 :00	Reserved	Reserved			

9.1.3 Serial ATA Index/Data Pair Superset Registers

All of these I/O registers are in the core well. They are exposed only when SCC is 01h (for example, IDE programming interface) and the controller is not in combined mode. These are Index/Data Pair registers that are used to access the SerialATA superset registers (SerialATA Status, SerialATA Control and SerialATA Error). The I/O space for these registers is allocated through SIDPBA. Locations with offset from 08h to 0Fh are reserved for future expansion. Software-write operations to the reserved locations shall have no effect while software-read operations to the reserved locations shall return 0.

Table 9-43. Serial ATA Index/Data Pair Superset Registers

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: SINDX—SATA* Index Register (B0:D31:F5)" on page 533	00000000h
04h	07h	"Offset 04h: SDATA—SATA* Index Data Register (B0:D31:F5)" on page 533	00000000h
04h	07h	"Offset 04h: PxSSTS—Serial ATA Status Register (B0:D31:F5)" on page 534	00000000h
04h	07h	"Offset 04h: PxSCTL—Serial ATA Control Register (B0:D31:F5)" on page 536	00000004h
04h	07h	"Offset 04h: PxSERR—Serial ATA Error Register (B0:D31:F5)" on page 538	00000000h

9.1.3.1 Offset 00h: SINDX—SATA* Index Register (B0:D31:F5)

Table 9-44. Offset 00h: SINDX—SATA* Index Register (B0:D31:F5)

Description:					
View: PCI	**BAR:** SIDPBA	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :08	PIDX	Port Index — This Index field is used to specify the port of the SATA* controller at which the port-specific SSTS, SCTL, and SERR registers are located. 00h = Primary Master (Port 4) 02h = Secondary Master (Port 5) All other values are Reserved.			RW
07 :00	RIDX	Register Index — This Index field is used to specify one out of three registers currently being indexed into. 00h = SSTS 01h = SCTL 02h = SERR All other values are Reserved			RW

9.1.3.2 Offset 04h: SDATA—SATA* Index Data Register (B0:D31:F5)

Table 9-45. Offset 04h: SDATA—SATA* Index Data Register (B0:D31:F5)

Description:					
View: PCI	**BAR** SIDPBA	**Bus:Device:Function:** B0:D31:F5		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DATA	Data — This Data register is a "window" through which data is read or written to the memory mapped registers. A read or write to this Data register triggers a corresponding read or write to the memory mapped register pointed to by the Index register. The Index register must be setup prior to the read or write to this Data register. A physical register is not actually implemented as the data is actually stored in the memory mapped registers. Since this is not a physical register, the "default" value is the same as the default value of the register pointed to by Index.			RW

9.1.3.3 Offset 04h: PxSSTS—Serial ATA Status Register (B0:D31:F5)

SDATA when SINDX.RIDX is 00h. This is a 32-bit register that conveys the current state of the interface and host. The PCH updates it continuously and asynchronously. When the PCH transmits a COMRESET to the device, this register is updated to its reset values.

Table 9-46. Offset 04h: PxSSTS—Serial ATA Status Register (B0:D31:F5) (Sheet 1 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR SIDPBA</td><td colspan="2">Bus:Device:Function: B0:D31 :F5</td><td>Offset Start: 04h
Offset End: 07h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="3">Default: 00000000h</td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :12</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>11 :08</td><td>IPM</td><td>Interface Power Management — Indicates the current interface state:

<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>Device not present or communication not established</td></tr><tr><td>1h</td><td>Interface in active state</td></tr><tr><td>2h</td><td>Interface in PARTIAL power management state</td></tr><tr><td>6h</td><td>Interface in SLUMBER power management state</td></tr></table>All other values reserved.</td><td></td><td></td><td>RO</td></tr>
<tr><td>07 :04</td><td>SPD</td><td>Current Interface Speed — Indicates the negotiated interface communication speed.

<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>Device not present or communication not established</td></tr><tr><td>1h</td><td>Generation 1 communication rate negotiated</td></tr><tr><td>2h</td><td>Generation 2 communication rate negotiated</td></tr></table>All other values reserved.

The PCH Supports Gen 1 communication rates (1.5 Gb/s) and Gen 2 rates
(3.0 Gb/s).</td><td></td><td></td><td>RO</td></tr>
</table>

Table 9-46. Offset 04h: PxSSTS—Serial ATA Status Register (B0:D31:F5) (Sheet 2 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR SIDPBA</td><td>Bus:Device:Function: B0:D31:F5</td><td>Offset Start: 04h
Offset End: 07h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: 00000000h</td><td></td><td colspan="2">Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td colspan="2">Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>03 :00</td><td>DET</td><td colspan="2">Device Detection — Indicates the interface device detection and Phy state:

<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>No device detected and Phy communication not established</td></tr><tr><td>1h</td><td>Device presence detected but Phy communication not established</td></tr><tr><td>3h</td><td>Device presence detected and Phy communication established</td></tr><tr><td>4h</td><td>Phy in offline mode as a result of the interface being disabled or running in a BIST loopback mode</td></tr></table>
All other values reserved.</td><td></td><td></td><td>RO</td></tr>
</table>

9.1.3.4 Offset 04h: PxSCTL—Serial ATA Control Register (B0:D31:F5)

SDATA when SINDX.RIDX is 01h. This is a 32-bit read-write register by which software controls SATA* capabilities. Writes to the SControl register result in an action being taken by the PCH or the interface. Reads from the register return the last value written to it.

Table 9-47. Offset 04h: PxSCTL—Serial ATA Control Register (B0:D31:F5) (Sheet 1 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR SIDPBA</td><td colspan="2">Bus:Device:Function: B0:D31 :F5</td><td>Offset Start: 04h
Offset End: 07h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: 00000004h</td><td colspan="2"></td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :20</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>19 :16</td><td>PMP</td><td>Port Multiplier Port — This field is not used by AHCI.</td><td></td><td></td><td>RO</td></tr>
<tr><td>15 :12</td><td>SPM</td><td>Select Power Management — This field is not used by AHCI.</td><td></td><td></td><td>RO</td></tr>
<tr><td>11 :08</td><td>IPM</td><td>Interface Power Management Transitions Allowed — Indicates which power states the PCH is allowed to transition to:
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>No interface restrictions</td></tr><tr><td>1h</td><td>Transitions to the PARTIAL state disabled</td></tr><tr><td>2h</td><td>Transitions to the SLUMBER state disabled</td></tr><tr><td>3h</td><td>Transitions to both PARTIAL and SLUMBER states disabled</td></tr></table>All other values reserved</td><td></td><td></td><td>RW</td></tr>
<tr><td>07 :04</td><td>SPD</td><td>Speed Allowed — Indicates the highest allowable speed of the interface. This speed is limited by the CAP.ISS (ABAR+00h:bit 23:20) field.
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>No speed negotiation restrictions</td></tr><tr><td>1h</td><td>Limit speed negotiation to Generation 1 communication rate</td></tr><tr><td>2h</td><td>Limit speed negotiation to Generation 2 communication rate</td></tr></table>All other values reserved.
The PCH Supports Gen 1 communication rates (1.5 Gb/s) and Gen 2 rates
(3.0 Gb/s).</td><td></td><td></td><td>RW</td></tr>
</table>

Table 9-47. Offset 04h: PxSCTL—Serial ATA Control Register (B0:D31:F5) (Sheet 2 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR SIDPBA</td><td>Bus:Device:Function: B0:D31:F5</td><td>Offset Start: 04h
Offset End: 07h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: 00000004h</td><td></td><td colspan="2">Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>03 :00</td><td>DET</td><td>Device Detection Initialization — Controls the PCH’s device detection and interface initialization.

<table><tr><th>Value</th><th>Description</th></tr><tr><td>0h</td><td>No device detection or initialization action requested</td></tr><tr><td>1h</td><td>Perform interface communication initialization sequence to establish communication. This is functionally equivalent to a hard reset and results in the interface being reset and communications re-initialized</td></tr><tr><td>4h</td><td>Disable the Serial ATA interface and put Phy in offline mode</td></tr></table>
All other values reserved.
When this field is written to a 1h, the PCH initiates COMRESET and starts the initialization process. When the initialization is complete, this field shall remain 1h until set to another value by software.
This field may only be changed to 1h or 4h when PxCMD.ST is 0. Changing this field while the PCH is running results in undefined behavior.</td><td></td><td></td><td>RW</td></tr>
</table>

9.1.3.5 Offset 04h: PxSERR—Serial ATA Error Register (B0:D31:F5)

SDATA when SINDx.RIDX is 02h.

Bits 26:16 of this register contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes. Bits 11:0 contain error information used by host software in determining the appropriate response to the error condition. If one or more of bits 11:8 of this register are set, the controller will stop the current transfer.

Table 9-48. Offset 04h: PxSERR—Serial ATA Error Register (B0:D31:F5) (Sheet 1 of 2)

Description:					
View: PCI	**BAR** SIDPBA	**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 04h **Offset End:** 07h		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved			
26	X	Exchanged — When set to one this bit indicates that a change in device presence has been detected since the last time this bit was cleared. This bit shall always be set to 1 anytime a COMINIT signal is received. This bit is reflected in the P0IS.PCS bit.			RWC
25	F	Unrecognized FIS Type — Indicates that one or more FISs were received by the Transport layer with good CRC, but had a type field that was not recognized.			RWC
24	T	Transport state transition error — Indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared.			RWC
23	T	Transport state transition error — Indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared.			RWC
22	H	Handshake — indicates that one or more R_ERR handshake response was received in response to frame transmission. Such errors may be the result of a CRC error detected by the recipient, a disparity or 8b/10b decoding error, or other error condition leading to a negative handshake on a transmitted frame.			RWC
21	C	CRC Error — Indicates that one or more CRC errors occurred with the Link Layer.			RWC
20	D	Disparity Error — This field is not used by AHCI.			RWC
19	B	10b to 8b Decode Error — Indicates that one or more 10b to 8b decoding errors occurred.			RWC
18	W	Comm Wake — Indicates that a Comm Wake signal was detected by the Phy.			RWC
17	I	Phy Internal Error — Indicates that the Phy detected some internal error.			RWC
16	N	PhyRdy Change — When set to 1 this bit indicates that the internal PhyRdy signal changed state since the last time this bit was cleared. In the PCH, this bit will be set when PhyRdy changes from a 0 -> 1 or a 1 -> 0. The state of this bit is then reflected in the PxIS.PRCS interrupt status bit and an interrupt will be generated if enabled. Software clears this bit by writing a 1 to it.			RWC
15 :12	Reserved	Reserved			

Table 9-48. Offset 04h: PxSERR—Serial ATA Error Register (B0:D31:F5) (Sheet 2 of 2)

Description:					
View: PCI	**BAR** SIDPBA		**Bus:Device:Function:** B0:D31:F5	**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	E	Internal Error — The SATA* controller failed due to a master or target abort when attempting to access system memory.			RWC
10	P	Protocol Error — A violation of the Serial ATA protocol was detected. Note: The PCH does not set this bit for all protocol violations that may occur on the SATA* link.			RWC
09	C	Persistent Communication or Data Integrity Error — A communication error that was not recovered occurred that is expected to be persistent. Persistent communications errors may arise from faulty interconnect with the device, from a device that has been removed or has failed, or a number of other causes.			RWC
07 :02	T	Transient Data Integrity Error — A data integrity error occurred that was not recovered by the interface.			RWC
01	Reserved	Reserved			O
00	M	Recovered Communications Error — Communications between the device and host was temporarily lost but was re-established. This can arise from a device temporarily being removed, from a temporary loss of Phy synchronization, or from other causes and may be derived from the PhyNRdy signal between the Phy and Link layers.			RWC

§ §

10.0 EHCI Controller Registers (B0:D29:F0)

10.1 USB* EHCI Configuration Registers (USB EHCI—B0:D29:F0)

Register address locations that are not shown in the following table should be treated as Reserved.

10.1.1 USB* EHCI PCI Register Address Map

Table 10-1. USB* EHCI PCI Register Address Map

View	BAR	BDF	Offset Start	Offset End	Register ID: Register Name	Default Value
PCI	Configuration	B0:D29:F0	00h	01h	"Offset 00h: VID: Vendor Identification Register (USB EHCI—B0:D29:F0)" on page 543	8086h
PCI	Configuration	B0:D29:F0	02h	03h	"Offset 02h: DID—Device Identification Register (USB EHCI—B0:D29:F0)" on page 543	2334h/23B4h & 2335h/23B5h[1]
PCI	Configuration	B0:D29:F0	04h	05h	"Offset 04h: PCI Command Register (USB EHCI—B0:D29:F0)" on page 544	0000h
PCI	Configuration	B0:D29:F0	06h	07h	"Offset 06h: PCI Status Register (USB EHCI—B0:D29:F0)" on page 545	0290h
PCI	Configuration	B0:D29:F0	08h	08h	"Offset 08h: RID—Revision Identification Register (USB EHCI—B0:D29:F0)" on page 546	See register description
PCI	Configuration	B0:D29:F0	09h	09h	"Offset 09h: Programming Interface Register (USB EHCI—B0:D29:F0)" on page 547	20h
PCI	Configuration	B0:D29:F0	0Ah	0Ah	"Offset 0Ah: Sub Class Code Register (USB EHCI—B0:D29:F0)" on page 547	03h
PCI	Configuration	B0:D29:F0	0Bh	0Bh	"Offset 0Bh: BCC—Base Class Code Register (USB EHCI—B0:D29:F0)" on page 547	0Ch
PCI	Configuration	B0:D29:F0	0Dh	0Dh	"Offset 0Dh: Primary Master Latency Timer Register (USB EHCI—B0:D29:F0)" on page 548	00h
PCI	Configuration	B0:D29:F0	0Eh	0Eh	"Offset 0Eh: Header Type Register (USB EHCI—B0:D29:F0)" on page 549	80h
PCI	Configuration	B0:D29:F0	10h	13h	"Offset 10h: Memory Base Address Register (USB EHCI—B0:D29:F0)" on page 549	00000000h
PCI	Configuration	B0:D29:F0	2Ch	2Dh	"Offset 2Ch: Subsystem Vendor ID Register (USB EHCI—B0:D29:F0)" on page 550	XXXXh
PCI	Configuration	B0:D29:F0	2Eh	2Fh	"Offset 2Eh: Subsystem ID Register (USB EHCI—B0:D29:F0)" on page 550	XXXXh
PCI	Configuration	B0:D29:F0	34h	34h	"Offset 34h: Capabilities Pointer Register (USB EHCI—B0:D29:F0)" on page 551	50h
PCI	Configuration	B0:D29:F0	3Ch	3Ch	"Offset 3Ch: Interrupt Line Register (USB EHCI—B0:D29:F0)" on page 551	00h

Table 10-1. USB* EHCI PCI Register Address Map

View	BAR	BDF	Offset Start	Offset End	Register ID: Register Name	Default Value
PCI	Configuration	B0:D29:F0	3Dh	3Dh	"Offset 3Dh: Interrupt Pin Register (USB EHCI—B0:D29:F0)" on page 551	See register description
PCI	Configuration	B0:D29:F0	50h	50h	"Offset 50h: PCI Power Management Capability ID Register (USB EHCI—B0:D29:F0)" on page 552	01h
PCI	Configuration	B0:D29:F0	51h	51h	"Offset 51h: Next Item Pointer #1 Register (USB EHCI—B0:D29:F0)" on page 552	58h
PCI	Configuration	B0:D29:F0	52h	53h	"Offset 52h: Power Management Capabilities Register (USB EHCI—B0:D29:F0)" on page 553	C9C2h
PCI	Configuration	B0:D29:F0	54h	55h	"Offset 54h: Power Management Control/Status Register (USB EHCI—B0:D29:F0)" on page 554	0000h
PCI	Configuration	B0:D29:F0	58h	58h	"Offset 58h: Debug Port Capability ID Register (USB EHCI—B0:D29:F0)" on page 555	0Ah
PCI	Configuration	B0:D29:F0	59h	59h	"Offset 59h: Next Item Pointer #2 Register (USB EHCI—B0:D29:F0)" on page 555	98h
PCI	Configuration	B0:D29:F0	5Ah	5Bh	"Offset 5Ah: Debug Port Base Offset Register (USB EHCI—B0:D29:F0)" on page 555	20A0h
PCI	Configuration	B0:D29:F0	60h	60h	"Offset 60h: USB* Release Number Register (USB EHCI—B0:D29:F0)" on page 556	20h
PCI	Configuration	B0:D29:F0	61h	61h	"Offset 61h: Frame Length Adjustment Register (USB EHCI—B0:D29:F0)" on page 557	20h
PCI	Configuration	B0:D29:F0	62h	63h	"Offset 62h: Port Wake Capability Register (USB EHCI—B0:D29:F0)" on page 558	01FFh
PCI	Configuration	B0:D29:F0	68h	6Bh	"Offset 68h: Legacy Support Extended Capability Register (USB EHCI—B0:D29:F0)" on page 559	00000001h
PCI	Configuration	B0:D29:F0	6Ch	6Fh	"Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0)" on page 560	00000000h
PCI	Configuration	B0:D29:F0	70h	73h	"Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register (USB EHCI—B0:D29:F0)" on page 563	00000000h
PCI	Configuration	B0:D29:F0	80h	80h	"Offset 80h: ACCESS_CNTL—Access Control Register (USB EHCI—B0:D29:F0)" on page 565	00h
PCI	Configuration	B0:D29:F0	84h	84h	"Offset 84h: EHCIIR1—EHCI Initialization Register 1 (USB EHCI—B0:D29:F0)" on page 565	01h
PCI	Configuration	B0:D29:F0	98h	98h	"Offset 98h: FLR_CID—Function Level Reset Capability ID (USB EHCI—B0:D29:F0)" on page 566	09h
PCI	Configuration	B0:D29:F0	99h	99h	"Offset 99h: FLR_NEXT—Function Level Reset Next Capability Pointer (USB EHCI—B0:D29:F0)" on page 566	00h
PCI	Configuration	B0:D29:F0	9Ah	9Bh	"Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0)" on page 567	2006h
PCI	Configuration	B0:D29:F0	9Ah	9Bh	"Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0)" on page 567	2006h
PCI	Configuration	B0:D29:F0	9Ch	9Ch	"Offset 9Ch: FLR_CTRL—Function Level Reset Control Register (USB EHCI—B0:D29:F0)" on page 568	00h
PCI	Configuration	B0:D29:F0	9Dh	9Dh	"Offset 9Dh: FLR_STS—Function Level Reset Status Register (USB EHCI—B0:D29:F0)" on page 568	00h
PCI	MBAR	B0:D29:F0	00h	00h	"Offset 00h: CAPLENGTH—Capability Registers Length (USB EHCI—B0:D29:F0)" on page 570	20h

Table 10-1. USB* EHCI PCI Register Address Map

View	BAR	BDF	Offset Start	Offset End	Register ID: Register Name	Default Value
PCI	MBAR	B0:D29:F0	02h	03h	"Offset 02h: HCIVERSION—Host Controller Interface Version Number Register (USB EHCI—B0:D29:F0)" on page 570	0100h
PCI	MBAR	B0:D29:F0	04h	07h	"Offset 04h: HCSPARAMS—Host Controller Structural Parameters (USB EHCI—B0:D29:F0)" on page 571	00204208h
PCI	MBAR	B0:D29:F0	08h	0Bh	"Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB EHCI—B0:D29:F0)" on page 572	00006881h
PCI	MBAR	B0:D29:F0	20h	23h	"Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0)" on page 574	00080000h
PCI	MBAR	B0:D29:F0	24h	27h	"Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0)" on page 577	00001000h
PCI	MBAR	B0:D29:F0	28h	2Bh	"Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)" on page 580	00000000h
PCI	MBAR	B0:D29:F0	2Ch	2Fh	"Offset 2Ch: FRINDEX—Frame Index Register (USB EHCI—B0:D29:F0)" on page 582	00000000h
PCI	MBAR	B0:D29:F0	30h	33h	"Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment Register (USB EHCI—B0:D29:F0)" on page 583	00000000h
PCI	MBAR	B0:D29:F0	34h	37h	"Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register (USB EHCI—B0:D29:F0)" on page 584	00000000h
PCI	MBAR	B0:D29:F0	38h	3Bh	"Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register (USB EHCI—B0:D29:F0)" on page 585	00000000h
PCI	MBAR	B0:D29:F0	60h	63h	"Offset 60h: CONFIGFLAG—Configure Flag Register (USB EHCI—B0:D29:F0)" on page 586	00000000h
PCI	MBAR	B0:D29:F0	64h	67h	"Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0)" on page 587	00003000h
PCI	MBAR	B0:D29:F0	A0h	A0h	"Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0)" on page 593	20h
PCI	MBAR	B0:D29:F0	A4h	A7h	"Offset A4h: USBPID—USB PIDs Register (USB EHCI—B0:D29:F0)" on page 595	00000000h
PCI	MBAR	B0:D29:F0	A8h	AFh	"Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register (USB EHCI—B0:D29:F0)" on page 596	0000000000000000h
PCI	MBAR	B0:D29:F0	B0h	B3h	"Offset B0h: CONFIG—Configuration Register (USB EHCI—B0:D29:F0)" on page 596	00007F01h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

Note: All configuration registers in this section are in the core well and reset by a core well reset and the D3-to-D0 warm reset, except as noted.

10.1.1.1 Offset 00h: Vendor Identification Register (USB EHCI—B0:D29:F0)

Table 10-2. Offset 00h: VID: Vendor Identification Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

10.1.1.2 Offset 02h: Device Identification Register (USB EHCI—B0:D29:F0)

Table 10-3. Offset 02h: DID—Device Identification Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2334h/23B4h & 2335h/23B5h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device ID — This is a 16-bit value assigned to the PCH USB* controller.		2334h/23B4h & 2335h/23B5h[1]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

10.1.1.3 Offset 04h: PCI Command Register (USB EHCI—B0:D29:F0)

Table 10-4. Offset 04h: PCI Command Register (Sheet 1 of 2) (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		00h	
10	INTx_Disable	Interrupt Disable: 0 = The function is capable of generating interrupts. 1 = The function can not generate its interrupt to the interrupt controller. The corresponding Interrupt Status bit (B0:D29:F0:06h, bit 3) is not affected by the interrupt enable.		0h	RW
09	FBE	Fast Back to Back Enable — Hardwired to 0.		0h	RO
08	SERR_EN	SERR# Enable: 0 = Disables EHC's capability to generate an SERR#. 1 = The Enhanced Host controller (EHC) is capable of generating (internally) SERR# in the following cases: - When it receive a completion status other than "successful" for one of its DMA initiated memory reads on DMI (and subsequently on its internal interface). - When it detects an address or command parity error and the Parity Error Response bit is set. - When it detects a data parity error (when the data is going into the EHC) and the Parity Error Response bit is set.		0h	RW
07	WCC	Wait Cycle Control — Hardwired to 0.		0h	RO
06	PER	Parity Error Response: 0 = The EHC is not checking for correct parity (on its internal interface). 1 = The EHC is checking for correct parity (on its internal interface) and halt operation when bad parity is detected during the data phase. NOTE: If the EHC detects bad parity on the address or command phases when the bit is set to 1, the host controller does not take the cycle. It halts the host controller (if currently not halted) and sets the Host System Error bit in the USBSTS register. This applies to both requests and completions from the system interface. This bit must be set in order for the parity errors to generate SERR#.		0h	RW
05	VPS	VGA Palette Snoop — Hardwired to 0.		0h	RO
04	PMWE	Postable Memory Write Enable — Hardwired to 0.		0h	RO
03	SCE	Special Cycle Enable — Hardwired to 0.		0h	RO

Table 10-4. Offset 04h: PCI Command Register (Sheet 2 of 2) (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BME	Bus Master Enable: 0 = Disables this functionality. 1 = Enables the PCH to act as a master on the PCI bus for USB* transfers.		1	RW
01	MSE	Memory Space Enable — This bit controls access to the USB* 2.0 Memory Space registers. 0 = Disables this functionality. 1 = Enables accesses to the USB* 2.0 registers. The Base Address register (B0:D29:F0:F0:10h) for USB* 2.0 should be programmed before this bit is set.		1	RW
00	IOSE	I/O Space Enable — Hardwired to 0.		1	RO

10.1.1.4 Offset 06h: PCI Status Register (USB EHCI—B0:D29:F0)

Table 10-5. Offset 06h: PCI Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0290h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the Parity Error Response bit (B0:D29:F0; Offset 04h bit 6) is not set.		0h	RWC
14	SSE	Signaled System Error: 0 = No SERR# signaled by the PCH. 1 = This bit is set by the PCH when it signals SERR# (internally). The SER_EN bit (bit 8 of the Command Register) must be 1 for this bit to be set.		0h	RWC
13	RMA	Received Master Abort: 0 = No master abort received. 1 = Set when the bridge receives a master abort status from the I/O data bus.		0h	RWC
12	RTA	Received Target Abort: 0 = No target abort received by EHC on memory access. 1 = This bit is set when EHC, as a master, receives a target abort status on a memory access. This is treated as a Host Error and halts the DMA engines. This event can optionally generate an SERR# by setting the SERR# Enable bit (B0:D29:F0:F0:04h, bit 8).			RWC
11	STA	Signaled Target Abort — This bit is used to indicate when the EHCI function responds to a cycle with a target abort. There is no reason for this to happen, so this bit is hardwired to 0.		0h	RO

Table 10-5. Offset 06h: PCI Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0290h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
10 :09	DEVT_STS	DEVSEL# Timing Status — This 2-bit field defines the timing for DEVSEL# assertion.		00h	RO
08	DPED	Master Data Parity Error Detected: 0 = No data parity error detected on USB*2.0 read completion packet. 1 = This bit is set by the PCH when a data parity error is detected on a USB* 2.0 read completion packet on the internal interface to the EHCI host controller and bit 6 of the Command register is set to 1.		0h	RWC
07	FB2BC	Fast Back to Back Capable — Hardwired to 1.		0h	RO
06	UDF	User Definable Features — Hardwired to 0.		0h	RO
05	66 MHz _CAP	66 MHz Capable — Hardwired to 0.		0h	RO
04	CAP_LIST	Capabilities List — Hardwired to 1 indicating that offset 34h contains a valid capabilities pointer.		1	RO
03	IS	Interrupt Status — This bit reflects the state of this function's interrupt at the input of the enable/disable logic. 0 = This bit will be 0 when the interrupt is deasserted. 1 = This bit is a 1 when the interrupt is asserted. The value reported in this bit is independent of the value in the Interrupt Enable bit.		0h	RO
02 :00	Reserved	Reserved			

10.1.1.5 Offset 08h: RID—Revision Identification Register (USB EHCI—B0:D29:F0)

Table 10-6. Offset 08h: RID—Revision Identification Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: This is an 8-bit value indicating the stepping of the USB* controller hardware.			RO

10.1.1.6 Offset 09h: Programming Interface Register (USB EHCI—B0:D29:F0)

Table 10-7. Offset 09h: Programming Interface Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Programming Interface — A value of 20h indicates that this USB* 2.0 host controller conforms to the EHCI Specification.			RO

10.1.1.7 Offset 0Ah: Sub Class Code Register (USB EHCI—B0:D29:F0)

Table 10-8. Offset 0Ah: Sub Class Code Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 03h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code: 03h = Universal serial bus host controller.			RO

10.1.1.8 Offset 0Bh: BCC—Base Class Code Register (USB EHCI—B0:D29:F0)

Table 10-9. Offset 0Bh: BCC—Base Class Code Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 0Ch			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code: 0Ch = Serial bus controller.			RO

10.1.1.9 Offset 0Dh: Primary Master Latency Timer Register (USB EHCI—B0:D29:F0)

Table 10-10. Offset 0Dh: Primary Master Latency Timer Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MLTC	Master Latency Timer Count: 00h = Hardwired. The SATA* controller is implemented internally, and is not arbitrated as a PCI device, so it does not need a Master Latency Timer.			RO

10.1.1.10 Offset 0Eh: Header Type Register (USB EHCI—B0:D29:F0)

Table 10-11. Offset 0Eh: Header Type Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 80h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Multi-Function Device — When set to '1' indicates this is a multifunction device: 0 = Single-function device 1 = Multi-function device. When RMH is enabled this bit defaults to 1, when RMH is disabled this bit defaults to 0.			RO
06 :00		Configuration Layout. Hardwired to 00h, which indicates the standard PCI configuration layout.			RO

10.1.1.11 Offset 10h: Memory Base Address Register (USB EHCI—B0:D29:F0)

Table 10-12. Offset 10h: Memory Base Address Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	BA	Base Address — Bits [31:10] correspond to memory address signals [31:10], respectively. This gives 1-KB of locatable memory space aligned to 1-KB boundaries.			RW
09 :04	Reserved	Reserved			
03	PREF	Prefetchable — Hardwired to 0 indicating that this range should not be prefetched.			RO
02 :01	TYPE	Type — Hardwired to 00b indicating that this range can be mapped anywhere within 32-bit address space.			RO
00	RTE	Resource Type Indicator — Hardwired to 0 indicating that the base address field in this register maps to memory space.			RO

10.1.1.12 Offset 2Ch: Subsystem Vendor ID Register (USB EHCI—B0:D29:F0)

Table 10-13. Offset 2Ch: Subsystem Vendor ID Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** XXXXh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	Subsystem Vendor ID — This register, in combination with the USB* 2.0 Subsystem ID register, enables the operating system to distinguish each subsystem from the others. Writes to this register are enabled when the WRT_RDONLY bit (B0:D29:F0:80h, bit 0) is set to 1.			RW

10.1.1.13 Offset 2Eh: Subsystem ID Register (USB EHCI—B0:D29:F0)

Table 10-14. Offset 2Eh: Subsystem ID Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** XXXXh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	Subsystem ID — BIOS sets the value in this register to identify the Subsystem ID. This register, in combination with the Subsystem Vendor ID register, enables the operating system to distinguish each subsystem from other(s). Writes to this register are enabled when the WRT_RDONLY bit (B0:D29:F0:80h, bit 0) is set to 1.			RW

10.1.1.14 Offset 34h: Capabilities Pointer Register (USB EHCI—B0:D29:F0)

Table 10-15. Offset 34h: Capabilities Pointer Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 50h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CAP_PTR	Capabilities Pointer — This register points to the starting offset of the USB* 2.0 capabilities ranges.			RO

10.1.1.15 Offset 3Ch: Interrupt Line Register (USB EHCI—B0:D29:F0)

Table 10-16. Offset 3Ch: Interrupt Line Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	INT_LN	Interrupt Line — This data is not used by the PCH. It is used as a scratchpad register to communicate to software the interrupt line to which that the interrupt pin is connected.			RW

10.1.1.16 Offset 3Dh: Interrupt Pin Register (USB EHCI—B0:D29:F0)

Table 10-17. Offset 3Dh: Interrupt Pin Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Interrupt Pin — This reflects the value of D29IP.E1IP (Chipset Config Registers:Offset 3108:bits 3:0). Bits 7:4 are always 0h			RO

10.1.1.17 Offset 50h: PCI Power Management Capability ID Register (USB EHCI—B0:D29:F0)

Table 10-18. Offset 50h: PCI Power Management Capability ID Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 50h **Offset End:** 50h	
Size: 8 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Power Management Capability ID — A value of 01h indicates that this is a PCI Power Management capabilities field.			RO

10.1.1.18 Offset 51h: Next Item Pointer #1 Register (USB EHCI—B0:D29:F0)

Table 10-19. Offset 51h: Next Item Pointer #1 Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 51h **Offset End:** 51h	
Size: 8 bit	**Default:** 58h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Next Item Pointer 1 Value — (special). This register defaults to 58h, which indicates that the next capability registers begin at configuration offset 58h. This register is writable when the WRT_RDONLY bit (B0:D29:F0:F0:80h, bit 0) is set. This allows BIOS to effectively hide the Debug Port capability registers, if necessary. This register should only be written during system initialization before the plug-and-play software has enabled any master-initiated traffic. Only values of 58h (Debug Port and FLR capabilities visible) and 98h (Debug Port invisible, next capability is FLR) are expected to be programmed in this register. ***Note:*** Register not reset by D3-to-D0 warm reset.			RW

10.1.1.19 Offset 52h: Power Management Capabilities Register (USB EHCI—B0:D29:F0)

Note: Normally, this register is read-only to report capabilities to the power management software. To report different power management capabilities, depending on the system in which the PCH is used, bits 15:11 and 8:6 in this register are writable when the WRT_RDONLY bit (B0:D29:F0:F0:80h, bit 0) is set. The value written to this register does not affect the hardware other than changing the value returned during a read.

- Reset: core well, but not D3-to-D0 warm reset.

Table 10-20. Offset 52h: Power Management Capabilities Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** C9C2h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	PME_SUP	PME Support — This 5-bit field indicates the power states in which the function may assert PME#. The PCH EHC does not support the D1 or D2 states. For all other states, the PCH EHC is capable of generating PME#. Software should never need to modify this field.			RW
10	D2_SUP	D2 Support: 0 = D2 State is not supported			RO
09	D1_SUP	D1 Support: 0 = D1 State is not supported			RO
08 :06	AUX_CUR	Auxiliary Current — The PCH EHC reports 375 mA maximum suspend well current required when in the $D3_{COLD}$ state.			RW
05	DSI	Device Specific Initialization — The PCH reports 0, indicating that no device-specific initialization is required.			RO
04	Reserved	Reserved			
03	PME_CLK	PME Clock — The PCH reports 0, indicating that no PCI clock is required to generate PME#.			RO
02 :00	VER	Version — The PCH reports 010b, indicating that it complies with Revision 1.1 of the PCI Power Management Specification.			RO

10.1.1.20 Offset 54h: Power Management Control/Status Register (USB EHCI—B0:D29:F0)

Reset (bits 15, 8): suspend well, and not D3-to-D0 warm reset or core well reset.

Table 10-21. Offset 54h: Power Management Control/Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 54h **Offset End:** 55h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		PME Status: 0 = Writing a 1 to this bit will clear it and cause the internal PME to deassert (if enabled). 1 = This bit is set when the PCH EHC would normally assert the PME# signal independent of the state of the PME_En bit. This bit must be explicitly cleared by the operating system each time the operating system is loaded. This bit is not reset by Function Level Reset.			RWC
14 :13		Data Scale — Hardwired to 00b indicating it does not support the associated Data register.			RO
12 :09		Data Select — Hardwired to 0000b indicating it does not support the associated Data register.			RO
08		PME Enable: 0 = Disable. 1 = Enables the PCH EHC to generate an internal PME signal when PME_Status is 1. This bit must be explicitly cleared by the operating system each time it is initially loaded. This bit is not reset by Function Level Reset.			RW
07 :02	Reserved	Reserved			
01 :00		Power State — This 2-bit field is used both to determine the current power state of EHC function and to set a new power state. The definition of the field values are: 00 = D0 state 11 = $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally; however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ state, the PCH must not accept accesses to the EHC memory range; but the configuration space must still be accessible. When not in the D0 state, the generation of the interrupt output is blocked. Specifically, the PIRQH is not asserted by the PCH when not in the D0 state. When software changes this value from the $D3_{HOT}$ state to the D0 state, an internal warm (soft) reset is generated, and software must re-initialize the function.			RW

10.1.1.21 Offset 58h: Debug Port Capability ID Register (USB EHCI—B0:D29:F0)

Table 10-22. Offset 58h: Debug Port Capability ID Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 58h **Offset End:** 58h	
Size: 8 bit	**Default:** 0Ah			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Debug Port Capability ID — Hardwired to 0Ah indicating that this is the start of a Debug Port Capability structure.			RO

10.1.1.22 Offset 59h: Next Item Pointer #2 Register (USB EHCI—B0:D29:F0)

Table 10-23. Offset 59h: Next Item Pointer #2 Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 59h **Offset End:** 59h	
Size: 8 bit	**Default:** 98h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Next Item Pointer 2 Capability — This register points to the next capability in the Function Level Reset capability structure.			RO

10.1.1.23 Offset 5Ah: Debug Port Base Offset Register (USB EHCI—B0:D29:F0)

Table 10-24. Offset 5Ah: Debug Port Base Offset Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 20A0h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13		BAR Number — Hardwired to 001b to indicate the memory BAR begins at offset 10h in the EHCI configuration space.			RO
12 :00		Debug Port Offset — Hardwired to 0A0h to indicate that the Debug Port registers begin at offset A0h in the EHCI memory range.			RO

10.1.1.24 Offset 60h: USB* Release Number Register (USB EHCI—B0:D29:F0)

Table 10-25. Offset 60h: USB* Release Number Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 60h **Offset End:** 60h	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		USB* Release Number — A value of 20h indicates that this controller follows *Universal Serial Bus (USB*) Specification, Revision 2.0*.			RO

10.1.1.25 Offset 61h: Frame Length Adjustment Register (USB EHCI—B0:D29:F0)

This feature is used to adjust any offset from the clock source that generates the clock that drives the SOF counter. When a new value is written into these six bits, the length of the frame is adjusted. Its initial programmed value is system dependent based on the accuracy of hardware USB* clock and is initialized by system BIOS. This register should only be modified when the HChalted bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 12) in the USB2.0_STS register is a 1. Changing value of this register while the host controller is operating yields undefined results. It should not be reprogrammed by USB* system software unless the default or BIOS programmed values are incorrect, or the system is restoring the register while returning from a suspended state.

These bits in suspend well and not reset by a D3-to-D0 warm rest or a core well reset.

Table 10-26. Offset 61h: Frame Length Adjustment Register (USB EHCI—B0:D29:F0)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D29:F0</td><td>Offset Start: 61h
Offset End: 61h</td></tr>
<tr><td colspan="2">Size: 8 bit</td><td colspan="2">Default: 20h</td><td></td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>07 :06</td><td>Reserved</td><td>Reserved — These bits are reserved for future use and should read as 00b.</td><td></td><td></td><td></td></tr>
<tr><td>05 :00</td><td></td><td>Frame Length Timing Value — Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives a SOF cycle time of 60000.
<table>
<tr><th>Frame Length (# 480 MHz Clocks) (decimal)</th><th>Frame Length Timing Value (this register) (decimal)</th></tr>
<tr><td>59488</td><td>0</td></tr>
<tr><td>59504</td><td>1</td></tr>
<tr><td>59520</td><td>2</td></tr>
<tr><td>—</td><td>—</td></tr>
<tr><td>59984</td><td>31</td></tr>
<tr><td>60000</td><td>32</td></tr>
<tr><td>—</td><td>—</td></tr>
<tr><td>60480</td><td>62</td></tr>
<tr><td>60496</td><td>63</td></tr>
</table></td><td></td><td></td><td>RW</td></tr>
</table>

10.1.1.26 Offset 62h: Port Wake Capability Register (USB EHCI—B0:D29:F0)

This register is in the suspend power well. The intended use of this register is to establish a policy about which ports are to be used for wake events. Bit positions 1–8(D29) in the mask correspond to a physical port implemented on the current EHCI controller. A 1 in a bit position indicates that a device connected below the port can be enabled as a wake-up device and the port may be enabled for disconnect/connect or overcurrent events as wake-up events. This is an information-only mask register. The bits in this register **do not** effect the actual operation of the EHCI host controller. The system-specific policy can be established by BIOS initializing this register to a system-specific value. System software uses the information in this register when enabling devices and ports for remote wake-up.

These bits are not reset by a D3-to-D0 warm rest or a core well reset.

Table 10-27. Offset 62h: Port Wake Capability Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29 :F0		**Offset Start:** 62h **Offset End:** 63h	
Size: 16 bit	**Default:** 01FFh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved			
08 :01		Port Wake Up Capability Mask — R/W. Bit positions 1 through 8 (Device 29) correspond to a physical port implemented on this host controller. For example, bit position 1 corresponds to port 1, bit position 2 corresponds to port 2, etc.			RW
00		Port Wake Implemented — R/W. A 1 in this bit indicates that this register is implemented to software.			RW

10.1.1.27 Offset 68h: Legacy Support Extended Capability Register (USB EHCI—B0:D29:F0)

These bits are not reset by a D3-to-D0 warm rest or a core well reset.

Table 10-28. Offset 68h: Legacy Support Extended Capability Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 68h **Offset End:** 6Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved — Hardwired to 00h			
24		HC OS Owned Semaphore — System software sets this bit to request ownership of the EHCI controller. Ownership is obtained when this bit reads as 1 and the HC BIOS Owned Semaphore bit reads as clear.			RW
23 :17	Reserved	Reserved — Hardwired to 00h			
16		HC BIOS Owned Semaphore — The BIOS sets this bit to establish ownership of the EHCI controller. System BIOS will clear this bit in response to a request for ownership of the EHCI controller by system software.			RW
15 :08		Next EHCI Capability Pointer — Hardwired to 00h to indicate that there are no EHCI Extended Capability structures in this device.			RO
07 :00		Capability ID — Hardwired to 01h to indicate that this EHCI Extended Capability is the Legacy Support Capability.			RO

10.1.1.28 Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0)

Table 10-29. Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 6Ch **Offset End:** 6Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		SMI on BAR — Software clears this bit by writing a 1 to it. 0 = Base Address Register (BAR) not written. 1 = This bit is set to 1 when the Base Address Register (BAR) is written.			RWC
30		SMI on PCI Command — Software clears this bit by writing a 1 to it. 0 = PCI Command (PCICMD) Register Not written. 1 = This bit is set to 1 when the PCI Command (PCICMD) Register is written.			RWC
29		SMI on OS Ownership Change — Software clears this bit by writing a 1 to it. 0 = No HC OS Owned Semaphore bit change. 1 = This bit is set to 1 when the HC OS Owned Semaphore bit in the LEG_EXT_CAP register (B0:D29:F0:F0:68h, bit 24) transitions from 1 to 0 or 0 to 1.			RWC
28 :22	Reserved	Reserved			
27		SMI on Async Advance — This bit is a shadow bit of the Interrupt on Async Advance bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 5) in the USB2.0_STS register. To clear this bit system software must write a 1 to the Interrupt on Async Advance bit in the USB2.0_STS register.			RO
26		SMI on Host System Error — This bit is a shadow bit of Host System Error bit in the USB2.0_STS register (B0:D29:F0:F0:CAPLENGTH + 24h, bit 4). To clear this bit system software must write a 1 to the Host System Error bit in the USB2.0_STS register.bit in the USB2.0_STS register.			RO
25		SMI on Frame List Rollover — This bit is a shadow bit of Frame List Rollover bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 3) in the USB2.0_STS register. To clear this bit system software must write a 1 to the Frame List Rollover bit in the USB2.0_STS register.			RO
24		SMI on Port Change Detect — This bit is a shadow bit of Port Change Detect bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 2) in the USB2.0_STS register. To clear this bit system software must write a 1 to the Port Change Detect bit in the USB2.0_STS register.			RO
23		SMI on USB* Error — This bit is a shadow bit of USB* Error Interrupt (USBERRINT) bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 1) in the USB2.0_STS register. To clear this bit system software must write a 1 to the USB* Error Interrupt bit in the USB2.0_STS register.			RO

Table 10-29. Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 6Ch **Offset End:** 6Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
22		SMI on USB* Complete — This bit is a shadow bit of USB* Interrupt (USBINT) bit (B0:D29:F0:F0:CAPLENGTH + 24h, bit 0) in the USB2.0_STS register. To clear this bit system software must write a 1 to the USB* Interrupt bit in the USB2.0_STS register.			RO
21		SMI on BAR Enable: 0 = Disable. 1 = Enable. When this bit is 1 and SMI on BAR (B0:D29:F0:F0:6Ch, bit 31) is 1, then the host controller will issue an SMI.			RW
20		SMI on PCI Command Enable: 0 = Disable. 1 = Enable. When this bit is 1 and SMI on PCI Command (B0:D29:F0:F0:6Ch, bit 30) is 1, then the host controller will issue an SMI.			RW
19		SMI on OS Ownership Enable: 0 = Disable. 1 = Enable. When this bit is a 1 AND the OS Ownership Change bit (B0:D29:F0:F0:6Ch, bit 29) is 1, the host controller will issue an SMI.			RW
18	Reserved	Reserved			
17		SMI on Async Advance Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on Async Advance bit (B0:D29:F0:6Ch, bit 21) is a 1, the host controller will issue an SMI immediately.			RW
16		SMI on Host System Error Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on Host System Error (B0:D29:F0:6Ch, bit 20) is a 1, the host controller will issue an SMI.			RW
15		SMI on Frame List Rollover Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on Frame List Rollover bit (B0:D29:F0:6Ch, bit 19) is a 1, the host controller will issue an SMI.			RW
14		SMI on Port Change Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on Port Change Detect bit (B0:D29:F0:6Ch, bit 18) is a 1, the host controller will issue an SMI.			RW
13		SMI on USB* Error Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on USB* Error bit (B0:D29:F0:6Ch, bit 17) is a 1, the host controller will issue an SMI immediately.			RW
12 :06		SMI on USB* Complete Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the SMI on USB* Complete bit (B0:D29:F0:6Ch, bit 16) is a 1, the host controller will issue an SMI immediately.			RW

Table 10-29. Offset 6Ch: Legacy Support Extended Control/Status Register (USB EHCI—B0:D29:F0) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 6Ch **Offset End:** 6Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05		SMI on BAR — Software clears this bit by writing a 1 to it. 0 = Base Address Register (BAR) not written. 1 = This bit is set to 1 when the Base Address Register (BAR) is written.			RWC
04		SMI on PCI Command — Software clears this bit by writing a 1 to it. 0 = PCI Command (PCICMD) Register Not written. 1 = This bit is set to 1 when the PCI Command (PCICMD) Register is written.			RWC
03		SMI on OS Ownership Change — Software clears this bit by writing a 1 to it. 0 = No HC OS Owned Semaphore bit change. 1 = This bit is set to 1 when the HC OS Owned Semaphore bit in the LEG_EXT_CAP register (B0:D29:F0:68h, bit 24) transitions from 1 to 0 or 0 to 1.			RWC
02	Reserved	Reserved			
01		SMI on Async Advance — This bit is a shadow bit of the Interrupt on Async Advance bit (D29:F0:CAPLENGTH + 24h, bit 5) in the USB2.0_STS register. To clear this bit system software must write a 1 to the Interrupt on Async Advance bit in the USB2.0_STS register.			RO
00		SMI on Host System Error — This bit is a shadow bit of Host System Error bit in the USB2.0_STS register (D29:F0:CAPLENGTH + 24h, bit 4). To clear this bit system software must write a 1 to the Host System Error bit in the USB2.0_STS register.			RO

10.1.1.29 Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register (USB EHCI—B0:D29:F0)

Table 10-30. Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register (USB EHCI—B0:D29:F0) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			
29 :22		SMI on PortOwner — Software clears these bits by writing a 1 to it. 0 = No Port Owner bit change. 1 = Bits 29:22, 27:22 correspond to the Port Owner bits for ports 0 (22) through 5 (27) or 7 (29). These bits are set to 1 when the associated Port Owner bits transition from 0 to 1 or 1 to 0.			RWC
21		SMI on PMCSR — Software clears these bits by writing a 1 to it. 0 = Power State bits Not modified. 1 = Software modified the Power State bits in the Power Management Control/Status (PMCSR) register (B0:D29:F0:54h).			RWC
20		SMI on Async — Software clears these bits by writing a 1 to it. 0 = No Async Schedule Enable bit change 1 = Async Schedule Enable bit transitioned from 1 to 0 or 0 to 1.			RWC
19		SMI on Periodic — Software clears this bit by writing a 1 it. 0 = No Periodic Schedule Enable bit change. 1 = Periodic Schedule Enable bit transitions from 1 to 0 or 0 to 1.			RWC
18		SMI on CF — Software clears this bit by writing a 1 it. 0 = No Configure Flag (CF) change. 1 = Configure Flag (CF) transitions from 1 to 0 or 0 to 1.			RWC
17		SMI on HCHalted — Software clears this bit by writing a 1 it. 0 = HCHalted did Not transition to 1 (as a result of the Run/Stop bit being cleared). 1 = HCHalted transitions to 1 (as a result of the Run/Stop bit being cleared).			RWC
16		SMI on HCReset — Software clears this bit by writing a 1 it. 0 = HCRESET did Not transitioned to 1. 1 = HCRESET transitioned to 1.			RWC
15 :14	Reserved	Reserved			
13 :06		SMI on PortOwner Enable: 0 = Disable. 1 = Enable. When any of these bits are 1 and the corresponding SMI on PortOwner bits are 1, then the host controller will issue an SMI. Unused ports should have their corresponding bits cleared.			RW
05		SMI on PMSCR Enable: 0 = Disable. 1 = Enable When this bit is 1 and SMI on PMSCR is 1, then the host controller will issue an SMI.			RW

Table 10-30. Offset 70h: SPECIAL_SMI—Intel Specific USB* 2.0 SMI Register (USB EHCI—B0:D29:F0) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Suspend	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
04		SMI on Async Enable: 0 = Disable. 1 = Enable When this bit is 1 and SMI on Async is 1, then the host controller will issue an SMI			RW
03		SMI on Periodic Enable: 0 = Disable. 1 = Enable When this bit is 1 and SMI on Periodic is 1, then the host controller will issue an SMI.			RW
02		SMI on CF Enable: 0 = Disable. 1 = Enable. When this bit is 1 and SMI on CF is 1, then the host controller will issue an SMI.			RW
01		SMI on HCHalted Enable: 0 = Disable. 1 = Enable. When this bit is a 1 and SMI on HCHalted is 1, then the host controller will issue an SMI.			RW
00		SMI on HCReset Enable: 0 = Disable. 1 = Enable. When this bit is a 1 and SMI on HCReset is 1, then host controller will issue an SMI.			RW

10.1.1.30 Offset 80h: ACCESS_CNTL—Access Control Register (USB EHCI—B0:D29:F0)

Table 10-31. Offset 80h: ACCESS_CNTL—Access Control Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 80h **Offset End:** 80h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved			
00	WRT_RDONLY	When set to 1, this bit enables a select group of normally read-only registers in the EHC function to be written by software. Registers that may only be written when this mode is entered are noted in the summary tables and detailed description as "Read/Write-Special". The registers fall into two categories: 1. System-configured parameters 2. Status bits			RW

10.1.1.31 Offset 84h: EHCIIR1—EHCI Initialization Register 1 (USB EHCI—B0:D29:F0)

Table 10-32. Offset 84h: EHCIIR1—EHCI Initialization Register 1 (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04		Pre-fetch Based Pause Disable: 0 = Pre-fetch Based Pause is enabled. 1 = Pre-fetch Based Pause is disabled.			RW
03 :00	Reserved	Reserved			

10.1.1.32 Offset 98h: FLR_CID—Function Level Reset Capability ID (USB EHCI—B0:D29:F0)

Table 10-33. Offset 98h: FLR_CID—Function Level Reset Capability ID (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 98h **Offset End:** 98h	
Size: 8 bit	**Default:** 09h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Capability ID: 13h = If FLRCSSEL = 0 09h (Vendor Specific Capability) = If FLRCSSEL = 1			RO

10.1.1.33 Offset 99h: FLR_NEXT—Function Level Reset Next Capability Pointer (USB EHCI—B0:D29:F0)

Table 10-34. Offset 99h: FLR_NEXT—Function Level Reset Next Capability Pointer (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 99h **Offset End:** 99h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		A value of 00h in this register indicates this is the last capability field			RO

10.1.1.34 Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0)

When FLRCSSEL = '0' this register is defined as follows:

Table 10-35. Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 9Ah **Offset End:** 9Bh	
Size: 16 bit	**Default:** 2006h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09		FLR Capability: 1 = Support for Function Level Reset (FLR).			RWO
08		TXP Capability: 1 = Support for Transactions Pending (TXP) bit. TXP must be supported if FLR is supported.			RWO
07 :00		Capability Length — This field indicates the # of bytes of this vendor specific capability as required by the PCI specification. It has the value of 06h for the FLR capability.			RO

When FLRCSSEL = '1' this register is defined as follows:

Table 10-36. Offset 9Ah: FLR_CLV—Function Level Reset Capability Length and Version (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 9Ah **Offset End:** 9Bh	
Size: 16 bit	**Default:** 2006h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12		Vendor Specific Capability ID — A value of 2h in this field identifies this capability as Function Level Reset.			RO
11 :08		Capability Version — This field indicates the version of the FLR capability.			RO
07 :00		Capability Length — This field indicates the # of bytes of this vendor specific capability as required by the PCI specification. It has the value of 06h for the FLR capability.			RO

10.1.1.35 Offset 9Ch: FLR_CTRL—Function Level Reset Control Register (USB EHCI—B0:D29:F0)

Table 10-37. Offset 9Ch: FLR_CTRL—Function Level Reset Control Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 9Ch **Offset End:** 9Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved			
00		Initiate FLR — This bit is used to initiate FLR transition. A write of '1' initiates FLR transition. Since hardware must not respond to any cycles until FLR completion, the value read by software from this bit is always '0'.			RW

10.1.1.36 Offset 9Dh: FLR_STS—Function Level Reset Status Register (USB EHCI—B0:D29:F0)

Table 10-38. Offset 9Dh: FLR_STS—Function Level Reset Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 9Dh **Offset End:** 9Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved			
00	TXP	Transactions Pending: 0 = Completions for all non-posted requests have been received. 1 = Controller has issued non-posted requests which have no bee completed.			RO

10.2 Memory-Mapped I/O Registers

The EHCI memory-mapped I/O space is composed of two sets of registers: Capability Registers and Operational Registers.

Note: The PCH EHCI controller will not accept memory transactions (neither reads nor writes) as a target that are locked transactions. The locked transactions should not be forwarded to PCI as the address space is known to be allocated to USB*.

- When the EHCI function is in the D3 PCI power state, accesses to the USB* 2.0 memory range are ignored and result a master abort. Similarly, if the Memory Space Enable (MSE) bit (B0:D29:F0:04h, bit 1) is not set in the Command register in configuration space, the memory range will not be decoded by the PCH enhanced host controller (EHC). If the MSE bit is not set, then the PCH must default to allowing any memory accesses for the range specified in the BAR to go to PCI. This is because the range may not be valid and, therefore, the cycle must be made available to any other targets that may be currently using that range.

10.2.1 Host Controller Capability Registers

These registers specify the limits, restrictions and capabilities of the host controller implementation. Within the host controller capability registers, only the structural parameters register is writable. These registers are implemented in the suspend well and is only reset by the standard suspend-well hardware reset, not by HCRESET or the D3-to-D0 reset.

Note: The EHCI controller does not support as a target memory transactions that are locked transactions. Attempting to access the EHCI controller Memory-Mapped I/O space using locked memory transactions will result in undefined behavior.

- When the USB*2.0 function is in the D3 PCI power state, accesses to the USB* 2.0 memory range are ignored and will result in a master abort Similarly, if the Memory Space Enable (MSE) bit is not set in the Command register in configuration space, the memory range will not be decoded by the Enhanced Host Controller (EHC). If the MSE bit is not set, then the EHC will not claim any memory accesses for the range specified in the BAR.

Table 10-39. Enhanced Host Controller Capability Registers

View	BAR	BDF	Offset Start	Offset End	Register ID: Register Name	Default Value
PCI	MBAR	B:29:0	00h	00h	"Offset 00h: CAPLENGTH—Capability Registers Length (USB EHCI—B0:D29:F0)" on page 570	20h
PCI	MBAR	B:29:0	02h	03h	"Offset 02h: HCIVERSION—Host Controller Interface Version Number Register (USB EHCI—B0:D29:F0)" on page 570	0100h
PCI	MBAR	B:29:0	04h	07h	"Offset 04h: HCSPARAMS—Host Controller Structural Parameters (USB EHCI—B0:D29:F0)" on page 571	00204208h
PCI	MBAR	B:29:0	08h	0Bh	"Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB EHCI—B0:D29:F0)" on page 572	00006881h

Note: "Read/Write Special" means that the register is normally read-only, but may be written when the WRT_RDONLY bit is set. Because these registers are expected to be programmed by BIOS during initialization, their contents must not get modified by HCRESET or D3-to-D0 internal reset.

10.2.1.1 Offset 00h: CAPLENGTH—Capability Registers Length Register (USB EHCI—B0:D29:F0)

Table 10-40. Offset 00h: CAPLENGTH—Capability Registers Length (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Capability Register Length Value — This register is used as an offset to add to the Memory Base Register (B0:D29:F0:Offset 10h) to find the beginning of the Operational Register Space. This field is hardwired to 20h indicating that the Operation Registers begin at offset 20h.			RO

10.2.1.2 Offset 02h: HCIVERSION—Host Controller Interface Version Number Register (USB EHCI—B0:D29:F0)

Table 10-41. Offset 02h: HCIVERSION—Host Controller Interface Version Number Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0100h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Host Controller Interface Version Number — This is a two-byte register containing a BCD encoding of the version number of interface that this host controller interface conforms.			RO

10.2.1.3 Offset 04h: HCSPARAMS—Host Controller Structural Parameters (USB EHCI—B0:D29:F0)

This register is writable when the WRT_RDONLY bit is set.

Table 10-42. Offset 04h: HCSPARAMS—Host Controller Structural Parameters (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00204208h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :20	DP_N	Debug Port Number — Hardwired to 2h indicating that the Debug Port is on the second lowest numbered port on the EHCI: EHCI#1: Port 1			RO
19 :16	Reserved	Reserved			
15 :12	N_CC	Number of Companion Controllers — This field indicates the number of companion controllers associated with this USB* EHCI host controller. BIOS must program this field to 0b to indicate that companion host controllers are not supported. Port ownership hand-off is not supported. Only high-speed devices are supported on the host controller root ports.			RW
11 :08	N_PCC	Number of Ports per Companion Controller — This field indicates the number of ports supported per companion host controller. This field is set to 0h to indicate no other companion support.			RO
07 :04	Reserved	Reserved. These bits are reserved and default to 0.			
03 :00	N_PORTS	This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 1h to Fh. A '0' in this field is undefined. For Integrated USB* 2.0 Rate Matching Hub Enabled: Each EHCI reports 2 ports by default. Port 0 assigned to the RMH and port 1 assigned as the debug port. When the KVM feature is enabled it will show up as Port2 on the EHCI, and BIOS would need to update this field to 3h.			RW

10.2.1.4 Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB* EHCI—B0:D29:F0)

Table 10-43. Offset 08h: HCCPARAMS—Host Controller Capability Parameters Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 08h **Offset End:** 0Bh		
Size: 32 bit	**Default:** 00006881h		**Power Well:**		

Bit Range	Bit Acronym	Bit Description	Sticky	Bit Reset Value	Bit Access
31 :18	Reserved	Reserved			
17	ASUC	Asynchronous Schedule Update Capability — There is no functionality associated with this bit.			RW
16	PSUC	Periodic Schedule Update Capability — This field is hardwired to 0b to indicate that the EHC hardware supports the Periodic Schedule Update Event Flag in the USB*2.0_CMD register.			RO
15 :08	EECP	EHCI Extended Capabilities Pointer — This field is hardwired to 68h, indicating that the EHCI capabilities list exists and begins at offset 68h in the PCI configuration space.			RO
07 :04		Isochronous Scheduling Threshold — This field indicates, relative to the current position of the executing host controller, where software can reliably update the isochronous schedule. When bit 7 is 0, the value of the least significant 3 bits indicates the number of micro-frames a host controller hold a set of isochronous data structures (one or more) before flushing the state. When bit 7 is a 1, then host software assumes the host controller may cache an isochronous data structure for an entire frame. See the EHCI specification for details on how software uses this information for scheduling isochronous transfers. This field is hardwired to 8h.			RO
03	Reserved	Reserved			
02		Asynchronous Schedule Park Capability — This bit is hardwired to 0 indicating that the host controller does not support this optional feature			RO
01		Programmable Frame List Flag: 0 = System software must use a frame list length of 1024 elements with this host controller. The USB*2.0_CMD register (B0:D29:F0:CAPLENGTH + 20h, bits 3:2) *Frame List Size* field is a read-only register and must be set to 0. 1 = System software can specify and use a smaller frame list and configure the host controller via the USB*2.0_CMD register *Frame List Size* field. The frame list must always be aligned on a 4K page boundary. This requirement ensures that the frame list is always physically contiguous.			RO
00		64-bit Addressing Capability — This field documents the addressing range capability of this implementation. The value of this field determines whether software should use the 32-bit or 64-bit data structures. This bit is hardwired to 1. The PCH supports 64 bit addressing only.			RO

10.2.2 Host Controller Operational Registers

This section defines the enhanced host controller operational registers. These registers are located after the capabilities registers. The operational register base must be dword-aligned and is calculated by adding the value in the first capabilities register (CAPLENGTH) to the base address of the enhanced host controller register address space (MEM_BASE). Since CAPLENGTH is always 20h, Table 10-44 accounts for this offset. All registers are 32 bits in length.

Table 10-44. Enhanced Host Controller Operational Register Address Map

View	BAR	BDF	Offset Start	Offset End	Register ID: Register Name	Default Value
PCI	MBAR	B:29:0	20h	23h	"Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0)" on page 574	00080000h
PCI	MBAR	B:29:0	24h	27h	"Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0)" on page 577	00001000h
PCI	MBAR	B:29:0	28h	2Bh	"Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)" on page 580	00000000h
PCI	MBAR	B:29:0	2Ch	2Fh	"Offset 2Ch: FRINDEX—Frame Index Register (USB EHCI—B0:D29:F0)" on page 582	00000000h
PCI	MBAR	B:29:0	30h	33h	"Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment Register (USB EHCI—B0:D29:F0)" on page 583	00000000h
PCI	MBAR	B:29:0	34h	37h	"Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register (USB EHCI—B0:D29:F0)" on page 584	00000000h
PCI	MBAR	B:29:0	38h	3Bh	"Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register (USB EHCI—B0:D29:F0)" on page 585	00000000h
PCI	MBAR	B:29:0	60h	63h	"Offset 60h: CONFIGFLAG—Configure Flag Register (USB EHCI—B0:D29:F0)" on page 586	00000000h
PCI	MBAR	B:29:0	64h	67h	"Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0)" on page 587	00003000h
PCI	MBAR	B:29:0	A0h	A0h	"Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0)" on page 593	20h
PCI	MBAR	B:29:0	A4h	A7h	"Offset A4h: USBPID—USB PIDs Register (USB EHCI—B0:D29:F0)" on page 595	00000000h
PCI	MBAR	B:29:0	A8h	AFh	"Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register (USB EHCI—B0:D29:F0)" on page 596	00000000 0000000h
PCI	MBAR	B:29:0	B0h	B3h	"Offset B0h: CONFIG—Configuration Register (USB EHCI—B0:D29:F0)" on page 596	00007F01h

Note: Software must read and write these registers using only dword accesses.These registers are divided into two sets. The first set at offsets MEM_BASE + 00:3Bh are implemented in the core power well. Unless otherwise noted, the core well registers are reset by the assertion of any of the following:

- Core well hardware reset
- HCRESET
- D3-to-D0 reset

The second set at offsets MEM_BASE + 60h to the end of the implemented register space are implemented in the Suspend power well. Unless otherwise noted, the suspend well registers are reset by the assertion of either of the following:

- Suspend well hardware reset
- HCRESET

10.2.2.1 Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0)

The Command Register indicates the command to be executed by the serial bus host controller. Writing to the register causes a command to be executed.

Table 10-45. Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 20h **Offset End:** 23h		
Size: 32 bit	**Default:** 00080000h		**Power Well:**		

Bit Range	Bit Acronym	Bit Description	Sticky	Bit Reset Value	Bit Access
31 :24	Reserved	Reserved			
23 :16		Interrupt Threshold Control — System software uses this field to select the maximum rate at which the host controller will issue interrupts. The only valid values are defined below. If software writes an invalid value to this register, the results are undefined. **Value** / **Maximum Interrupt Interval** 00h — Reserved 01h — 1 micro-frame 02h — 2 micro-frames 04h — 4 micro-frames 08h — 8 micro-frames (default, equates to 1 ms) 10h — 16 micro-frames (2 ms) 20h — 32 micro-frames (4 ms) 40h — 64 micro-frames (8 ms)			RW
15 :14	Reserved	Reserved			
13	ASC	Asynch Schedule Update — There is no functionality associated with this bit.			RW
12		Periodic Schedule Prefetch Enable — This bit is used by software to enable the host controller to prefetch the periodic schedule even in C0. 0 = Prefetch based pause enabled only when not in C0. 1 = Prefetch based pause enable in C0. Once software has written a 1b to this bit to enable periodic schedule prefetching, it must disable prefecthing by writing a 0b to this bit whenever periodic schedule updates are about to begin. Software should continue to dynamically disable and re-enable the prefetcher surrounding any updates to the periodic scheduler (for example, until the host controller has been reset via a HCRESET).			RW
11 :08		Unimplemented Asynchronous Park Mode Bits — Hardwired to 000b indicating the host controller does not support this optional feature.			RO

Table 10-45. Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00080000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Light Host Controller Reset — Hardwired to 0. The PCH does not implement this optional reset.			RO
06		Interrupt on Async Advance Doorbell — This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. 0 = The host controller sets this bit to a 0 after it has set the Interrupt on Async Advance status bit (B0:D29:F0:CAPLENGTH + 24h, bit 5) in the USB2.0_STS register to a 1. 1 = Software must write a 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USB2.0_STS register. If the *Interrupt on Async Advance Enable* bit in the USB*2.0_INTR register (B0:D29:F0:CAPLENGTH + 28h, bit 5) is a 1 then the host controller will assert an interrupt at the next interrupt threshold. See the EHCI specification for operational details. Software should not write a 1 to this bit when the asynchronous schedule is inactive. Doing so will yield undefined results.			RW
05		Asynchronous Schedule Enable — This bit controls whether the host controller skips processing the Asynchronous Schedule. 0 = Do not process the Asynchronous Schedule 1 = Use the ASYNCLISTADDR register to access the Asynchronous Schedule.			RW
04		Periodic Schedule Enable — This bit controls whether the host controller skips processing the Periodic Schedule. 0 = Do not process the Periodic Schedule 1 = Use the PERIODICLISTBASE register to access the Periodic Schedule.			RW
03 :02		Frame List Size — The PCH hardwires this field to 00b because it only supports the 1024-element frame list size.			RO
01	HCRESET	Host Controller Reset — This control bit used by software to reset the host controller. The effects of this on root hub registers are similar to a Chip Hardware Reset (for example, RSMRST# assertion and PWROK deassertion on the PCH). When software writes a 1 to this bit, the host controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB* is immediately terminated. A USB* reset is not driven on downstream ports. PCI configuration registers and Host controller capability registers are not effected by this reset. All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s), with the side effects described in the EHCI spec. Software must re-initialize the host controller in order to return the host controller to an operational state. This bit is set to 0 by the host controller when the reset process is complete. Software cannot terminate the reset process early by writing a 0 to this register. Software should not set this bit to a 1 when the HCHalted bit (B0:D29:F0:CAPLENGTH + 24h, bit 12) in the USB2.0_STS register is a 0. Attempting to reset an actively running host controller will result in undefined behavior. This reset me be used to leave EHCI port test modes.			RW

Table 10-45. Offset 20h: USB*2.0_CMD—USB 2.0 Command Register (USB EHCI—B0:D29:F0) (Sheet 3 of 3)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: MBAR</td><td>Bus:Device:Function: B0:D29:F0</td><td>Offset Start: 20h
Offset End: 23h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: 00080000h</td><td colspan="2"></td><td>Power Well:</td></tr>
<tr><th>Bit Range</th><th>Bit Acronym</th><th>Bit Description</th><th>Sticky</th><th>Bit Reset Value</th><th>Bit Access</th></tr>
<tr><td>00</td><td>RS</td><td>Run/Stop:
0 = Stop (default)
1 = Run. When set to a 1, the Host controller proceeds with execution of the schedule. The Host controller continues execution as long as this bit is set. When this bit is set to 0, the Host controller completes the current transaction on the USB* and then halts. The HCHalted bit in the USB2.0_STS register indicates when the Host controller has finished the transaction and has entered the stopped state.
Software should not write a 1 to this field unless the host controller is in the Halted state (for example, HCHalted in the USBSTS register is a 1). The Halted bit is cleared immediately when the Run bit is set.
The following table explains how the different combinations of Run and Halted should be interpreted:
<table><tr><th>Run/Stop</th><th>Halted</th><th>Interpretation</th></tr><tr><td>0b</td><td>0b</td><td>In the process of halting</td></tr><tr><td>0b</td><td>1b</td><td>Halted</td></tr><tr><td>1b</td><td>0b</td><td>Running</td></tr><tr><td>1b</td><td>1b</td><td>Invalid - the HCHalted bit clears immediately</td></tr></table>Memory read cycles initiated by the EHC that receive any status other than Successful will result in this bit being cleared.</td><td></td><td></td><td>RW</td></tr>
</table>

10.2.2.2 Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0)

This register indicates pending interrupts and various states of the Host controller. The status resulting from a transaction on the serial bus is not indicated in this register. See the Interrupts description in section 4 of the EHCI specification for additional information concerning USB* 2.0 interrupt conditions.

For the writable bits, software must write a 1 to clear bits that are set. Writing a 0 has no effect.

Table 10-46. Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00001000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15		Asynchronous Schedule Status — This bit reports the current real status of the Asynchronous Schedule. 0 = Status of the Asynchronous Schedule is disabled. (Default) 1 = Status of the Asynchronous Schedule is enabled. The Host controller is not required to *immediately* disable or enable the Asynchronous Schedule when software transitions the *Asynchronous Schedule Enable* bit (B0:D29:F0:CAPLENGTH + 20h, bit 5) in the USB2.0_CMD register. When this bit and the *Asynchronous Schedule Enable* bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).			RO
14		Periodic Schedule Status — This bit reports the current real status of the Periodic Schedule. 0 = Status of the Periodic Schedule is disabled. (Default) 1 = Status of the Periodic Schedule is enabled. The Host controller is not required to *immediately* disable or enable the Periodic Schedule when software transitions the *Periodic Schedule Enable* bit (B0:D29:F0:CAPLENGTH + 20h, bit 4) in the USB2.0_CMD register. When this bit and the *Periodic Schedule Enable* bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).			RO
13		Reclamation — This read-only status bit is used to detect an empty asynchronous schedule. The operational model and valid transitions for this bit are described in Section 4 of the EHCI Specification.			RO
12		HCHalted: 0 = This bit is a 0 when the Run/Stop bit is a 1. 1 = The Host controller sets this bit to 1 after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host controller hardware (for example, internal error). (Default)			RO
11 :06	Reserved	Reserved			

Table 10-46. Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00001000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05		Interrupt on Async Advance — System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a 1 to the *Interrupt on Async Advance Doorbell* bit (B0:D29:F0:CAPLENGTH + 20h, bit 6) in the USB2.0_CMD register. This bit indicates the assertion of that interrupt source.			RWC
04		Host System Error: 0 = No serious error occurred during a host system access involving the Host controller module 1 = The Host controller sets this bit to 1 when a serious error occurs during a host system access involving the Host controller module. A hardware interrupt is generated to the system. Memory read cycles initiated by the EHC that receive any status other than Successful will result in this bit being set. When this error occurs, the Host controller clears the Run/Stop bit in the USB2.0_CMDregister (B0:D29:F0:CAPLENGTH + 20h, bit 0) to prevent further execution of the scheduled TDs. A hardware interrupt is generated to the system (if enabled in the Interrupt Enable Register).			RWC
03		Frame List Rollover: 0 = No *Frame List Index* rollover from its maximum value to 0. 1 = The Host controller sets this bit to a 1 when the *Frame List Index* rolls over from its maximum value to 0. Since the PCH only supports the 1024-entry Frame List Size, the *Frame List Index* rolls over every time FRNUM13 toggles.			RWC

Table 10-46. Offset 24h: USB2.0_STS—USB 2.0 Status Register (USB EHCI—B0:D29:F0) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00001000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02		Port Change Detect — This bit is allowed to be maintained in the Auxiliary power well. Alternatively, it is also acceptable that on a D3 to D0 transition of the EHCI HC device, this bit is loaded with the OR of all of the PORTSC change bits (including: Force port resume, overcurrent change, enable/disable change and connect status change). Regardless of the implementation, when this bit is readable (for example, in the D0 state), it must provide a valid view of the Port Status registers. 0 = No change bit transition from a 0 to 1 or No Force Port Resume bit transition from 0 to 1 as a result of a J-K transition detected on a suspended port. 1 = The Host controller sets this bit to 1 when any port for which the *Port Owner* bit is set to 0 has a change bit transition from a 0 to 1 or a Force Port Resume bit transition from 0 to 1 as a result of a J-K transition detected on a suspended port.			RWC
01	USBERRINT	USB* Error Interrupt: 0 = No error condition. 1 = The Host controller sets this bit to 1 when completion of a USB* transaction results in an error condition (for example, error counter underflow). If the TD on which the error interrupt occurred also had its IOC bit set, both this bit and Bit 0 are set. See the EHCI specification for a list of the USB* errors that will result in this interrupt being asserted.			RWC
00	USBINT	USB* Interrupt: 0 = No completion of a USB* transaction whose Transfer Descriptor had its IOC bit set. No short packet is detected. 1 = The Host controller sets this bit to 1 when the cause of an interrupt is a completion of a USB* transaction whose Transfer Descriptor had its IOC bit set. The Host controller also sets this bit to 1 when a short packet is detected (actual number of bytes received was less than the expected number of bytes).			RWC

10.2.2.3 Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)

This register enables and disables reporting of the corresponding interrupt to the software. When a bit is set and the corresponding interrupt is active, an interrupt is generated to the host. Interrupt sources that are disabled in this register still appear in the USB2.0_STS Register to allow the software to poll for events. Each interrupt enable bit description indicates whether or not it is dependent on the interrupt threshold mechanism (See Section 4 of the EHCI specification).

Table 10-47. Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 28h **Offset End:** 2Bh		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :06	Reserved	Reserved			
05		Interrupt on Async Advance Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the Interrupt on Async Advance bit (B0:D29:F0:CAPLENGTH + 24h, bit 5) in the USB2.0_STS register is a 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.			RW
04		Host System Error Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the Host System Error Status bit (B0:D29:F0:CAPLENGTH + 24h, bit 4) in the USB2.0_STS register is a 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.			RW
03		Frame List Rollover Enable:. 0 = Disable. 1 = Enable. When this bit is a 1, and the Frame List Rollover bit (B0:D29:F0:CAPLENGTH + 24h, bit 3) in the USB2.0_STS register is a 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit.			RW

Table 10-47. Offset 28h: USB2.0_INTR—USB 2.0 Interrupt Enable Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 28h **Offset End:** 2Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02		Port Change Interrupt Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the Port Change Detect bit (B0:D29:F0:CAPLENGTH + 24h, bit 2) in the USB2.0_STS register is a 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Change Detect bit.			RW
01		USB* Error Interrupt Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the USBERRINT bit (B0:D29:F0:CAPLENGTH + 24h, bit 1) in the USB2.0_STS register is a 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software by clearing the USBERRINT bit in the USB2.0_STS register.			RW
00		USB* Interrupt Enable: 0 = Disable. 1 = Enable. When this bit is a 1, and the USBINT bit (B0:D29:F0:CAPLENGTH + 24h, bit 0) in the USB2.0_STS register is a 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software by clearing the USBINT bit in the USB2.0_STS register.			RW

10.2.2.4 Offset 2Ch: FRINDEX—Frame Index Register (USB EHCI—B0:D29:F0)

The SOF frame number value for the bus SOF token is derived or alternatively managed from this register. See Section 4 of the EHCI specification for a detailed explanation of the SOF value management requirements on the host controller. The value of FRINDEX must be within 125 µs (1 micro-frame) ahead of the SOF token value. The SOF value may be implemented as an 11-bit shadow register. For this discussion, this shadow register is 11 bits and is named SOFV. SOFV updates every 8 micro-frames (1 millisecond). An example implementation to achieve this behavior is to increment SOFV each time the FRINDEX[2:0] increments from 0 to 1.

Software must use the value of FRINDEX to derive the current micro-frame number, both for high-speed isochronous scheduling purposes and to provide the **get** micro-frame number function required to client drivers. Therefore, the value of FRINDEX and the value of SOFV must be kept consistent if chip is reset or software writes to FRINDEX. Writes to FRINDEX must also **write-through** FRINDEX[13:3] to SOFV[10:0]. To keep the update as simple as possible, software should never write a FRINDEX value where the three least significant bits are 111b or 000b.

Note: sThis register is used by the host controller to index into the periodic frame list. The register updates every 125 microseconds (once each micro-frame). Bits [12:3] are used to select a particular entry in the Periodic Frame List during periodic schedule execution. The number of bits used for the index is fixed at 10 for the PCH since it only supports 1024-entry frame lists. This register must be written as a dword. Word and byte writes produce undefined results. This register cannot be written unless the Host controller is in the Halted state as indicated by the *HCHalted* bit (B0:D29:F0:CAPLENGTH + 24h, bit 12). A write to this register while the Run/Stop bit (B0:D29:F0:CAPLENGTH + 20h, bit 0) is set to a 1 (USB2.0_CMD register) produces undefined results. Writes to this register also effect the SOF value. See Section 4 of the EHCI specification for details.

Table 10-48. Offset 2Ch: FRINDEX—Frame Index Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 2Ch **Offset End:** 2Fh		
Size: 32 bit	**Default:** 00000000h		**Power Well:**		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13 :00		Frame List Current Index/Frame Number — The value in this register increments at the end of each time frame (for example, micro-frame). Bits [12:3] are used for the Frame List current index. This means that each location of the frame list is accessed 8 times (frames or micro-frames) before moving to the next index.			RW

10.2.2.5 Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment (USB EHCI—B0:D29:F0) Register

This 32-bit register corresponds to the most significant address bits [63:32] for all EHCI data structures. Since the PCH hardwires the 64-bit Addressing Capability field in HCCPARAMS to 1, then this register is used with the link pointers to construct 64-bit addresses to EHCI control data structures. This register is concatenated with the link pointer from either the PERIODICLISTBASE, ASYNCLISTADDR, or any control data structure link field to construct a 64-bit address. This register allows the host software to locate all control data structures within the same 4 GB memory segment.

Table 10-49. Offset 30h: CTRLDSSEGMENT—Control Data Structure Segment Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 30h **Offset End:** 33h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12		Upper Address[63:44] — Hardwired to 0s. The PCH EHC is only capable of generating addresses up to 16 terabytes (44 bits of address).			RO
11 :00		Upper Address[43:32] — This 12-bit field corresponds to address bits 43:32 when forming a control data structure address.			RW

10.2.2.6 Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register (USB EHCI—B0:D29:F0)

This 32-bit register contains the beginning address of the Periodic Frame List in the system memory. Since the PCH host controller operates in 64-bit mode (as indicated by the 1 in the 64-bit Addressing Capability field in the HCCSPARAMS register) (offset 08h, bit 0), then the most significant 32 bits of every control data structure address comes from the CTRLDSSEGMENT register. HCD loads this register prior to starting the schedule execution by the host controller. The memory structure referenced by this physical memory pointer is assumed to be 4-Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host controller to step through the Periodic Frame List in sequence.

Table 10-50. Offset 34h: PERIODICLISTBASE—Periodic Frame List Base Address Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 34h **Offset End:** 37h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	BA (LOW)	Base Address — These bits correspond to memory address signals [31:12], respectively.			RW
11 :00	Reserved	Reserved			

10.2.2.7 Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register (USB EHCI—B0:D29:F0)

This 32-bit register contains the address of the next asynchronous queue head to be executed. Since the PCH host controller operates in 64-bit mode (as indicated by a 1 in 64-bit Addressing Capability field in the HCCPARAMS register) (offset 08h, bit 0), then the most significant 32 bits of every control data structure address comes from the CTRLDSSEGMENT register (offset 08h). Bits [4:0] of this register cannot be modified by system software and will always return 0s when read. The memory structure referenced by this physical memory pointer is assumed to be 32-byte aligned.

Table 10-51. Offset 38h: ASYNCLISTADDR—Current Asynchronous List Address Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 38h **Offset End:** 3Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	LPL	Link Pointer Low — These bits correspond to memory address signals [31:5], respectively. This field may only reference a Queue Head (QH).			RW
04 :00	Reserved	Reserved			

10.2.2.8 Offset 60h: CONFIGFLAG—Configure Flag Register (USB EHCI—B0:D29:F0)

This register is in the suspend power well. It is only reset by hardware when the suspend power is initially applied or in response to a host controller reset.

Table 10-52. Offset 60h: CONFIGFLAG—Configure Flag Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 60h **Offset End:** 63h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved			
00	CF	Configure Flag — Host software sets this bit as the last action in its process of configuring the Host controller. This bit controls the default port-routing control logic. Bit values and side-effects are listed below. See section 4 of the EHCI spec for operation details. 0 = Port routing control logic default-routes each port to the UHCIs (default). 1 = Port routing control logic default-routes all ports to this host controller.			RW

10.2.2.9 Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0)

Note: The EHCI RMH is always enabled.

Offset: Port 0 RMH: MEM_BASE + 64h–67h
Port 1 Debug Port: MEM_BASE + 68–6Bh
Port 2 USB* redirect (if enabled): MEM_BASE + 6C–6Fh

Note: This register is associated with the upstream ports of the EHCI controller and does not represent downstream hub ports. USB* Hub class commands must be used to determine RMH port status and enable test modes. See Chapter 11 of the USB* Specification, Revision 2.0 for more details. Rate Matching Hub wake capabilities can be configured by the RMHWKCTL Register (RCBA+35B0h) located in the Chipset Configuration chapter.

A host controller must implement one or more port registers. Software uses the N_Port information from the Structural Parameters Register to determine how many ports need to be serviced. All ports have the structure defined below. Software must not write to unreported Port Status and Control Registers.

This register is in the suspend power well. It is only reset by hardware when the suspend power is initially applied or in response to a host controller reset. The initial conditions of a port are:

- No device connected
- Port disabled.

When a device is attached, the port state transitions to the attached state and system software will process this as with any status change notification. See Section 4 of the EHCI specification for operational requirements for how change events interact with port suspend mode.

Table 10-53. Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) (Sheet 1 of 5)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00003000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved	Reserved			
22	WKOC_E	Wake on Overcurrent Enable: 0 = Disable. (Default) 1= Enable. Writing this bit to a 1 enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the overcurrent Active bit (bit 4 of this register) is set.			RW
21	WKDSCNNT_E	Wake on Disconnect Enable: 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1 enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the Current Connect Status changes from connected to disconnected (for example, bit 0 of this register changes from 1 to 0).			RW

Table 10-53. Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) (Sheet 2 of 5)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00003000h			**Power Well:**	

Bit Range	Bit Acronym	Bit Description	Sticky	Bit Reset Value	Bit Access
20	WKCNNT_E	Wake on Connect Enable: 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1 enables the setting of the PME Status bit in the Power Management Control/Status Register (offset 54, bit 15) when the Current Connect Status changes from disconnected to connected (for example, bit 0 of this register changes from 0 to 1).			RW
19 :16		Port Test Control — When this field is 0s, the port is NOT operating in a test mode. A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value. The encoding of the test mode bits are (0110b – 1111b are reserved): **Value** / **Maximum Interrupt Interval** 0000b — Test mode not enabled (default) 0001b — Test J_STATE 0010b — Test K_STATE 0011b — Test SE0_NAK 0100b — Test Packet 0101b — FORCE_ENABLE See USB* Specification Revision 2.0, Chapter 7 for details on each test mode.			RW
15 :14	Reserved	Reserved			
13		Port Owner — This bit unconditionally goes to a 0 when the Configured Flag bit in the USB2.0_CMD register makes a 0 to 1 transition. System software uses this field to release ownership of the port to a selected host controller (in the event that the attached device is not a high-speed device). Software writes a 1 to this bit when the attached device is not a high-speed device. A 1 in this bit means that a companion host controller owns and controls the port. See Section 4 of the EHCI Specification for operational details.			RW
12	PP	Port Power — Read-only with a value of 1. This indicates that the port does have power.			RO
11 :10		Line Status— These bits reflect the current logical levels of the D+ (bit 11) and D– (bit 10) signal lines. These bits are used for detection of low-speed USB* devices prior to the port reset and enable sequence. This field is valid only when the port enable bit is 0 and the current connect status bit is set to a 1. 00 = SE0 10 = J-state 01 = K-state 11 = Undefined			RO
09	Reserved	Reserved			

Table 10-53. Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) (Sheet 3 of 5)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00003000h			**Power Well:**	

Bit Range	Bit Acronym	Bit Description	Sticky	Bit Reset Value	Bit Access
08		Port Reset — When software writes a 1 to this bit (from a 0), the bus reset sequence as defined in the USB* Specification, Revision 2.0 is started. Software writes a 0 to this bit to terminate the bus reset sequence. Software must keep this bit at a 1 long enough to ensure the reset sequence completes as specified in the USB* Specification, Revision 2.0. 1 = Port is in Reset. 0 = Port is not in Reset. When software writes a 0 to this bit, there may be a delay before the bit status changes to a 0. The bit status will not read as a 0 until after the reset has completed. If the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (for example, set the *Port Enable* bit to a 1). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from 0 to 1. For example: if the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state within 2 ms of software writing this bit to a 0. The HCHalted bit (B0:D29:F0:CAPLENGTH + 24h, bit 12) in the USB2.0_STS register should be a 0 before software attempts to use this bit. The host controller may hold Port Reset asserted to a 1 when the HCHalted bit is a 1. This bit is 0 if Port Power is 0 System software should not attempt to reset a port if the *HCHalted* bit in the USB2.0_STS register is a 1. Doing so will result in undefined behavior.			RW
07		Suspend: 0 = Port not in suspend state.(Default) 1 = Port in suspend state. Port Enabled Bit and Suspend bit of this register define the port states as follows: **Port Enabled** / **Suspend** / **Port State** 0 / X / Disabled 1 / 0 / Enabled 1 / 1 / Suspend When in suspend state, downstream propagation of data is blocked on this port, except for port reset. The bit status does not change until the port is suspended and that there may be a delay in suspending a port depending on the activity on the port. The host controller will unconditionally set this bit to a 0 when software sets the *Force Port Resume* bit to a 0 (from a 1). A write of 0 to this bit is ignored by the host controller. If host software sets this bit to a 1 when the port is not enabled (for example, Port enabled bit is a 0) the results are undefined.			RW

Table 10-53. Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) (Sheet 4 of 5)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00003000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06		Force Port Resume: 0 = No resume (K-state) detected/driven on port. (Default) 1 = Resume detected/driven on port. Software sets this bit to a 1 to drive resume signaling. The Host controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a 1 because a J-to-K transition is detected, the Port Change Detect bit (B0:D29:F0:CAPLENGTH + 24h, bit 2) in the USB2.0_STS register is also set to a 1. If software sets this bit to a 1, the host controller must not set the Port Change Detect bit. When the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB* Specification, Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this bit remains a 1. Software must appropriately time the Resume and set this bit to a 0 when the appropriate amount of time has elapsed. Writing a 0 (from 1) causes the port to return to high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a 1 until the port has switched to the high-speed idle.			RW
05		Overcurrent Change — The functionality of this bit is not dependent upon the port owner. Software clears this bit by writing a 1 to it. 0 = No change. (Default) 1 = There is a change to Overcurrent Active.			RWC
04		Overcurrent Active: 0 = This port does not have an overcurrent condition. (Default) 1 = This port currently has an overcurrent condition. This bit will automatically transition from 1 to 0 when the over current condition is removed. The PCH automatically disables the port when the overcurrent active bit is 1.			RO
03		Port Enable/Disable Change — For the root hub, this bit gets set to a 1 only when a port is disabled due to the appropriate conditions existing at the EOF2 point (See Chapter 11 of the USB* Specification for the definition of a port error). This bit is not set due to the Disabled-to-Enabled transition, nor due to a disconnect. Software clears this bit by writing a 1 to it. 0 = No change in status. (Default). 1 = Port enabled/disabled status has changed.			RWC
02		Port Enabled/Disabled — Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a 1 to this bit. Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by host software. The bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. 0 = Disable 1 = Enable (Default)			RW

Table 10-53. Offset 64h: PORTSC—Port N Status and Control Register (USB EHCI—B0:D29:F0) (Sheet 5 of 5)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00003000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
01		Connect Status Change — This bit indicates a change has occurred in the port's Current Connect Status. Software sets this bit to 0 by writing a 1 to it. 0 = No change (Default). 1 = Change in Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. For example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be "setting" an already-set bit (for example, the bit will remain set).			RWC
00		Current Connect Status — This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change bit (Bit 1) to be set. 0 = No device is present. (Default) 1 = Device is present on port.			RO

10.2.3 USB* 2.0-Based Debug Port Registers

The Debug port's registers are located in the same memory area, defined by the Base Address Register (MEM_BASE), as the standard EHCI registers. The base offset for the debug port registers (A0h) is declared in the Debug Port Base Offset Capability Register at Configuration offset 5Ah (B0:D29:F0:offset 5Ah). The specific EHCI port that supports this debug capability (Port 1 for B0:D29:F0) is indicated by a 4-bit field (bits 20–23) in the HCSPARAMS register of the EHCI controller. The address map of the Debug Port registers is shown in Table 10-54.

Table 10-54. Debug Port Register Address Map

MEM_BASE + Offset	Mnemonic	Register Name	Default	Type
A0–A3h	CNTL_STS	Control/Status	00000000h	R/W, R/WC, RO
A4–A7h	USBPID	USB* PIDs	00000000h	R/W, RO
A8–AFh	DATABUF[7:0]	Data Buffer (Bytes 7:0)	00000000 00000000h	R/W
B0–B3h	CONFIG	Configuration	00007F01h	R/W

Notes:
1. All of these registers are implemented in the core well and reset by PLTRST#, EHC HCRESET, and a EHC D3-to-D0 transition.
2. The hardware associated with this register provides no checks to ensure that software programs the interface correctly. How the hardware behaves when programmed improperly is undefined.

10.2.3.1 Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0)

Note: Software should do Read-Modify-Write operations to this register to preserve the contents of bits not being modified. This include Reserved bits.

- To preserve the usage of Reserved bits in the future, software should always write the same value read from the bit until it is defined. Reserved bits will always return 0 when read.

Table 10-55. Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** A0h **Offset End:** A0h	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved			
30	OWNER_CNT	0 = Ownership of the debug port is NOT forced to the EHCI controller (Default) 1 = Ownership of the debug port is forced to the EHCI controller (for example, immediately taken away from the companion Classic USB* Host controller) If the port was already owned by the EHCI controller, then setting this bit has no effect. This bit overrides all of the ownership-related bits in the standard EHCI registers.			RW
29	Reserved	Reserved			
28	ENABLED_CNT	0 = Software can clear this by writing a 0 to it. The hardware clears this bit for the same conditions where the Port Enable/Disable Change bit (in the PORTSC register) is set. (Default) 1 = Debug port is enabled for operation. Software can directly set this bit if the port is already enabled in the associated PORTSC register (this is enforced by the hardware).			RW
27 :17	Reserved	Reserved			
16	DONE_STS	Software can clear this by writing a 1 to it. 0 = Request Not complete 1 = Set by hardware to indicate that the request is complete.			RWC
15 :12	LINK_ID_STS	This field identifies the link interface. 0h = Hardwired. Indicates that it is a USB* Debug Port.			RO
11	Reserved	Reserved			
10	IN_USE_CNT	Set by software to indicate that the port is in use. Cleared by software to indicate that the port is free and may be used by other software. This bit is cleared after reset. (This bit has no affect on hardware.)			RW
09 :07	EXCEPTION_STS	This field indicates the exception when the ERROR_GOOD#_STS bit is set. This field should be ignored if the ERROR_GOOD#_STS bit is 0. 000 =No Error. (Default) **Note:** This should not be seen since this field should only be checked if there is an error. 001 =Transaction error: Indicates the USB* 2.0 transaction had an error (CRC, bad PID, timeout, etc.) 010 =Hardware error. Request was attempted (or in progress) when port was suspended or reset. All Other combinations are reserved			RO

Table 10-55. Offset A0h: CNTL_STS—Control/Status Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR		**Bus:Device:Function:** B0:D29:F0	**Offset Start:** A0h **Offset End:** A0h	
Size: 8 bit	**Default:** 20h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	ERROR_GOOD#_STS	0 = Hardware clears this bit to 0 after the proper completion of a read or write. (Default) 1 = Error has occurred. Details on the nature of the error are provided in the Exception field.			RO
05	GO_CNT	0 = Hardware clears this bit when hardware sets the DONE_STS bit. (Default) 1 = Causes hardware to perform a read or write request. Writing a 1 to this bit when it is already set may result in undefined behavior.			RW
04	WRITE_READ#_CNT	Software clears this bit to indicate that the current request is a read. Software sets this bit to indicate that the current request is a write. 0 = Read (Default) 1 = Write			RW
03 :00	DATA_LEN_CNT	This field is used to indicate the size of the data to be transferred. default = 0h. For write operations, this field is set by software to indicate to the hardware how many bytes of data in Data Buffer are to be transferred to the console. A value of 0h indicates that a zero-length packet should be sent. A value of 1–8 indicates 1–8 bytes are to be transferred. Values 9–Fh are invalid and how hardware behaves if used is undefined. For read operations, this field is set by hardware to indicate to software how many bytes in Data Buffer are valid in response to a read operation. A value of 0h indicates that a zero length packet was returned and the state of Data Buffer is not defined. A value of 1–8 indicates 1–8 bytes were received. Hardware is not allowed to return values 9–Fh. The transferring of data always starts with byte 0 in the data area and moves toward byte 7 until the transfer size is reached.			RW

10.2.3.2 Offset A4h: USBPID—USB PIDs Register (USB EHCI—B0:D29:F0)

This Dword register is used to communicate PID information between the USB* debug driver and the USB* debug port. The debug port uses some of these fields to generate USB* packets, and uses other fields to return PID information to the USB* debug driver.

Table 10-56. Offset A4h: USBPID—USB PIDs Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23 :16	RECEIVED_PID_STS[23:16]	Hardware updates this field with the received PID for transactions in either direction. When the controller is writing data, this field is updated with the handshake PID that is received from the device. When the host controller is reading data, this field is updated with the data packet PID (if the device sent data), or the handshake PID (if the device NAKs the request). This field is valid when the hardware clears the GO_DONE#_CNT bit.			RO
15 :08	SEND_PID_CNT[15:8]	Hardware sends this PID to begin the data packet when sending data to USB* (for example, WRITE_READ#_CNT is asserted). Software typically sets this field to either DATA0 or DATA1 PID values.			RW
07 :00	TOKEN_PID_CNT[7:0]	Hardware sends this PID as the Token PID for each USB* transaction. Software typically sets this field to either IN, OUT, or SETUP PID values.			RW

10.2.3.3 Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register (USB EHCI—B0:D29:F0)

This register can be accessed as eight separate 8-bit registers or two separate 32-bit registers.

Table 10-57. Offset A8h: DATABUF[7:0]—Data Buffer Bytes[7:0] Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** A8h **Offset End:** AFh	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :00		DATABUFFER[63:0] — his field is the 8 bytes of the data buffer. Bits 7:0 correspond to least significant byte (byte 0). Bits 63:56 correspond to the most significant byte (byte 7). The bytes in the Data Buffer must be written with data before software initiates a write request. For a read request, the Data Buffer contains valid data when DONE_STS bit (offset A0, bit 16) is cleared by the hardware, ERROR_GOOD#_STS (offset A0, bit 6) is cleared by the hardware, and the DATA_LENGTH_CNT field (offset A0, bits 3:0) indicates the number of bytes that are valid.			RW

10.2.3.4 Offset B0h: CONFIG—Configuration Register (USB EHCI—B0:D29:F0)

Table 10-58. Offset B0h: CONFIG—Configuration Register (USB EHCI—B0:D29:F0)

Description:					
View: PCI	**BAR:** MBAR	**Bus:Device:Function:** B0:D29:F0		**Offset Start:** B0h **Offset End:** B3h	
Size: 8 bit	**Default:** 00007F01h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :15	Reserved	Reserved			
14 :08	USB_ADDRESS_CNF	This 7-bit field identifies the USB* device address used by the controller for all Token PID generation. (Default = 7Fh)			RW
07 :04	Reserved	Reserved			
03 :00	USB_EndPoint_CNF	This 4-bit field identifies the EndPoint used by the controller for all Token PID generation. (Default = 1h)			RW

§ §

11.0 SMBus Controller Registers (B0:D31:F3)

11.1 PCI Configuration Registers (SMBus—B0:D31:F3)

11.1.1 SMBus Controller PCI Register Address Map

Table 11-1. SMBus Controller PCI Register Address Map

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: Vendor Identification Register (SMBus—B0:D31:F3)" on page 598	8086h
02h	03h	"Offset 02h: Device Identification Register (SMBus—B0:D31:F3)" on page 598	2330h/23B0h[1]
04h	05h	"Offset 04h: PCI Command Register (SMBus—B0:D31:F3)" on page 599	0000h
06h	07h	"Offset 06h: PCI Status Register (SMBus—B0:D31:F3)" on page 600	0280h
08h	08h	"Offset 08h: Revision Identification Register (SMBus—B0:D31:F3)" on page 601	See register description
09h	09h	"Offset 09h: Programming Interface Register (SMBus—B0:D31:F3)" on page 601	00h
0Ah	09h	"Offset 0Ah: Sub Class Code Register (SMBus—B0:D31:F3)" on page 601	05h
0Bh	0Bh	"Offset 0Bh: Base Class Code Register (SMBus—B0:D31:F3)" on page 602	0Ch
10h	13h	"Offset 10h: SMBus Memory Base Address 0 (SMBus—B0:D31:F3)" on page 602	00000004h
14h	17h	"Offset 14h: SMBus Memory Base Address 1 (SMBus—B0:D31:F3)" on page 603	00000000h
20h	23h	"Offset 20h: SMBus Base Address Register (SMBus—B0:D31:F3)" on page 603	00000001h
2Ch	2Dh	"Offset 2Ch: Subsystem Vendor Identification Register (SMBus—B0:D31:F3)" on page 604	0000h
2Eh	2Fh	"Offset 2Eh: Subsystem Identification Register (SMBus—B0:D31:F3)" on page 604	0000h
3Ch	3Ch	"Offset 3Ch: Interrupt Line Register (SMBus—B0:D31:F3)" on page 605	00h
3Dh	3Dh	"Offset 3Dh: Interrupt Pin Register (SMBus—B0:D31:F3)" on page 605	See register description
40h	40h	"Offset 40h: Host Configuration Register (SMBus—B0:D31:F3)" on page 606	00h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

Note: Registers that are not shown should be treated as Reserved.

11.1.1.1 Offset 00h: Vendor Identification Register (SMBus—B0:D31:F3)

Table 11-2. Offset 00h: Vendor Identification Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

11.1.1.2 Offset 02h: Device Identification Register (SMBus—B0:D31:F3)

Table 11-3. Offset 02h: Device Identification Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2330h/23B0h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device Identification: Indicates the device number assigned by the SIG.		2330h/23B0h[a]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

11.1.1.3 Offset 04h: PCI Command Register (SMBus—B0:D31:F3)

Table 11-4. Offset 04h: PCI Command Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		00h	
10		Interrupt Disable — This disables pin-based INTx# interrupts. This bit has no effect on MSI operation. 0 = Internal INTx# messages are generated if there is an interrupt and MSI is not enabled. 1 = Internal INTx# messages will not be generated.		0h	RW
09	FBE	Fast Back to Back Enable — Reserved as 0.		0h	RO
08	SERR_EN	SERR# Enable — Reserved as 0.		0h	RW
07	WCC	Wait Cycle Control — Reserved as 0.		0h	RO
06	PER	Parity Error Response: 0 = Disabled. SATA* controller will not generate PERR# when a data parity error is detected. 1 = Enabled. SATA* controller will generate PERR# when a data parity error is detected.		0h	RW
05	VPS	VGA Palette Snoop — Reserved as 0.		0h	RO
04	PMWE	Postable Memory Write Enable — Reserved as 0.		0h	RO
03	SCE	Special Cycle Enable — Reserved as 0.		0h	RO
02	BME	Bus Master Enable — This bit controls the PCH's ability to act as a PCI master for IDE Bus Master transfers. This bit does not impact the generation of completions for split transaction commands.		1	RW
01	MSE	Memory Space Enable — Controls access to the SATA* controller's target memory space (for AHCI). This bit is RO '0' when not in AHCI mode.		1	RW/RO
00	IOSE	I/O Space Enable — This bit controls access to the I/O space registers. 0 = Disables access to the Legacy or Native IDE ports (both Primary and Secondary) as well as the Bus Master I/O registers. 1 = Enable. The Base Address register for the Bus Master registers should be programmed before this bit is set.		1	RW

11.1.1.4 Offset 06h: PCI Status Register (SMBus—B0:D31:F3)

Table 11-5. Offset 06h: PCI Status Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0280h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the Parity Error Response bit (D30:F0:04 bit 6) is not set.			RWC
14	SSE	Signaled System Error — Set when the LPC bridge signals a system error to the internal SERR# logic.			RWC
13	RMA	Received Master Abort: 0 = No master abort received. 1 = Set when the bridge receives a master abort status from the I/O data bus.			RWC
12	RTA	Received Target Abort — Hardwired to 0.			RC
11	STA	Signaled Target Abort — Reserved as 0.			RO
10 :09	DEV_STS	DEVSEL# Timing Status: 01 = Hardwired; Controls the device select time for the SATA* controller's PCI interface.			RO
08	DPED	Data Parity Error Detected — For PCH, this bit can only be set on read completions received from the bus when there is a parity error. SATA* controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set.			RWC
07	FB2BC	Fast Back to Back Capable — Hardwired to 1.			RO
06	UDF	User Definable Features — Hardwired to 0.			RO
05	66MHZ_CAP	66 MHz Capable — Hardwired to 0.			RO
04	CAP_LIST	Capabilities List — Hardwired to 0 because there are no capability list structures in this function			RO
03	INTS	Interrupt Status — This bit indicates that an interrupt is pending. It is independent from the state of the Interrupt Enable bit in the PCI Command register.			RO
02 :00	Reserved	Reserved			

11.1.1.5 Offset 08h: Revision Identification Register (SMBus—B0:D31:F3)

Table 11-6. Offset 08h: Revision Identification Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID — This is an 8-bit value indicating the stepping of the SMBus controller hardware.		Variable	RO

11.1.1.6 Offset 09h: Programming Interface Register (SMBus—B0:D31:F3)

Table 11-7. Offset 09h: Programming Interface Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Reserved	Reserved		00h	

11.1.1.7 Offset 0Ah: Sub Class Code Register (SMBus—B0:D31:F3)

Table 11-8. Offset 0Ah: Sub Class Code Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Ah **Offset End:** 09h	
Size: 8 bit	**Default:** 05h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code — SMBus serial controller		05h	RO

11.1.1.8 Offset 0Bh: Base Class Code Register (SMBus—B0:D31:F3)

Table 11-9. Offset 0Bh: Base Class Code Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 0Ch			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code — Serial controller		0Ch	RO

11.1.1.9 Offset 10h: SMBus Memory Base Address 0 (SMBus—B0:D31:F3)

Table 11-10. Offset 10h: SMBus Memory Base Address 0 (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000004h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	BA	Base Address — Provides the 32 byte system memory base address for the PCH SMB logic.			RW
07 :04	Reserved	Reserved			
03	PREF	Prefetchable — Hardwired to 0. Indicates that SMBMBAR is not pre-fetchable.			RO
02 :01	ADDRNG	Address Range — Indicates that this SMBMBAR can be located anywhere in 64 bit address space. Hardwired to 10b.			RO
00		Memory Space Indicator — This read-only bit always is 0, indicating that the SMB logic is Memory mapped.			RO

11.1.1.10 Offset 14h: SMBus Memory Base Address 1 (SMBus—B0:D31:F3)

Table 11-11. Offset 14h: SMBus Memory Base Address 1 (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BA	Base Address — Provides bits 63-32 system memory base address for the PCH SMB logic.			RW

11.1.1.11 Offset 20h: SMBus Base Address Register (SMBus—B0:D31:F3)

Table 11-12. Offset 20h: SMBus Base Address Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :05	BA	Base Address — This field provides the base address of the I/O space (16 consecutive I/O locations).			RW
04 :01	Reserved	Reserved			
00		IO Space Indicator — Hardwired to 1 indicating that the SMB logic is I/O mapped.			RO

11.1.1.12 Offset 2Ch: Subsystem Vendor Identification Register (SMBus—B0:D31:F2/F4)

Table 11-13. Offset 2Ch: Subsystem Vendor Identification Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	Subsystem Vendor ID — The SVID register, in combination with the Subsystem ID (SID) register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads to this register is the same as that which was written by BIOS into the IDE SVID register. Software can write to this register only once per core well reset. Writes should be done as a single 16-bit cycle.			RO

11.1.1.13 Offset 2Eh: Subsystem Identification Register (SMBus—B0:D31:F3)

Table 11-14. Offset 2Eh: Subsystem Identification Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	Subsystem ID — The SID register, in combination with the SVID register, enables the operating system (OS) to distinguish subsystems from each other. The value returned by reads to this register is the same as that which was written by BIOS into the IDE SID register. Software can write to this register only once per core well reset. Writes should be done as a single 16-bit cycle.			RWO

11.1.1.14 Offset 3Ch: Interrupt Line Register (SMBus—B0:D31:F3)

Table 11-15. Offset 3Ch: Interrupt Line Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	INT_LN	Interrupt Line — This data is not used by the PCH. It is to communicate to software the interrupt line that the interrupt pin is connected to PIRQB#.			RW

11.1.1.15 Offset 3Dh: Interrupt Pin Register (SMBus—B0:D31:F3)

Table 11-16. Offset 3Dh: Interrupt Pin Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	INT_PN	Interrupt PIN — This reflects the value of D31IP.SMIP in chipset configuration space.			RO

11.1.1.16 Offset 40h: Host Configuration Register (SMBus—B0:D31:F3)

Table 11-17. Offset 40h: Host Configuration Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 40h **Offset End:** 40h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved			
03	SSRESET	Soft SMBus Reset: 0 = The HW will reset this bit to 0 when SMBus reset operation is completed. 1 = The SMBus state machine and logic in the PCH is reset.			RW
02	I^2C_EN	0 = SMBus behavior. 1 = The PCH is enabled to communicate with I^2C devices. This will change the formatting of some commands.			RW
01	MB_SMI_EN	0 = SMBus interrupts will not generate an SMI# 1 = Any source of an SMB interrupt will instead be routed to generate an SMI#. This bit needs to be set for SMBALERT# to be enabled.			RW
00	HST_EN	SMBus Host Enable: 0 = Disable the SMBus Host controller 1 = Enable. The SMB Host controller interface is enabled to execute commands. The INTREN bit (offset SMBASE + 02h, bit 0) needs to be enabled for the SMB Host controller to interrupt or SMI#. The SMB Host controller will not respond to any new requests until all interrupt requests have been cleared.			RW

11.2 SMBus I/O and Memory Mapped I/O Registers

The following SMBus registers can be accessed through I/O BAR or Memory BAR registers in PCI configuration space. The offsets are the same for both I/O and Memory Mapped I/O registers.

11.2.1 SMBus Registers

Table 11-18. SMBus Bus Registers

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: Host Status Register (SMBus—B0:D31:F3)" on page 608	00h
02h	02h	"Offset 02h: Host Control Register (SMBus—B0:D31:F3)" on page 610	00h
03h	03h	"Offset 03h: Host Command Register (SMBus—B0:D31:F3)" on page 612	00h
04h	04h	"Offset 04h: Transmit Slave Address Register (SMBus—B0:D31:F3)" on page 612	00h
05h	05h	"Offset 05h: Host Data 0 Register (SMBus—B0:D31:F3)" on page 613	00h
06h	06h	"Offset 06h: Host Data 1 Register (SMBus—B0:D31:F3)" on page 613	00h
07h	07h	"Offset 07h: Host Block Data Byte Register (SMBus—B0:D31:F3)" on page 614	00h
08h	08h	"Offset 08h: Packet Error Check (PEC) Register (SMBus—B0:D31:F3)" on page 615	00h
09h	09h	"Offset 09h: Receive Slave Address Register (SMBus—B0:D31:F3)" on page 615	44h
0Ah	0Bh	"Offset 0Ah: Receive Slave Data Register (SMBus—B0:D31:F3)" on page 616	0000h
0Ch	0Ch	"Offset 0Ch: Auxiliary Status Register (SMBus—B0:D31:F3)" on page 616	00h
0Dh	0Dh	"Offset 0Dh: Auxiliary Control Register (SMBus—B0:D31:F3)" on page 617	00h
0Eh	0Eh	"Offset 0Eh: SMLink Pin Control Register (SMBus—B0:D31:F3)" on page 618	See register description
0Fh	0Fh	"Offset 0Fh: SMBus Pin Control Register (SMBus—B0:D31:F3)" on page 619	See register description
10h	10h	"Offset 10h: Slave Status Register (SMBus—B0:D31:F3)" on page 620	00h
11h	11h	"Offset 11h: Slave Command Register (SMBus—B0:D31:F3)" on page 621	00h
14h	14h	"Offset 14h: Notify Device Address Register (SMBus—B0:D31:F3)" on page 622	00h
16h	16h	"Offset 16h: Notify Data Low Byte Register (SMBus—B0:D31:F3)" on page 622	00h
17h	17h	"Offset 17h: Notify Data High Byte Register (SMBus—B0:D31:F3)" on page 623	00h

11.2.1.1 Offset 00h: Host Status Register (SMBus—B0:D31:F3)

All status bits are set by hardware and cleared by the software writing a one to the particular bit position. Writing a 0 to any bit position has no effect.

Table 11-19. Offset 00h: Host Status Register (SMBus—B0:D31:F3) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 00h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	DS	Byte Done Status: 0 = Software can clear this by writing a 1 to it. 1 = Host controller received a byte (for Block Read commands) or if it has completed transmission of a byte (for Block Write commands) when the 32-byte buffer is not being used. This bit will be set, even on the last byte of the transfer. This bit is not set when transmission is due to the LAN interface heartbeat. This bit has no meaning for block transfers when the 32-byte buffer is enabled. When the last byte of a block message is received, the host controller will set this bit. However, it will not immediately set the INTR bit (bit 1 in this register). When the interrupt handler clears the DS bit, the message is considered complete, and the host controller will then set the INTR bit (and generate another interrupt). Thus, for a block message of n bytes, the PCH will generate n+1 interrupts. The interrupt handler needs to be implemented to handle these cases. When not using the 32 Byte Buffer, hardware will drive the SMBCLK signal low when the DS bit is set until SW clears the bit. This includes the last byte of a transfer. Software must clear the DS bit before it can clear the BUSY bit.				RWC
06	INUSE_STS	This bit is used as semaphore among various independent software threads that may need to use the PCH's SMBus logic, and has no other effect on hardware. 0 = After a full PCI reset, a read to this bit returns a 0. 1 = After the first read, subsequent reads will return a 1. A write of a 1 to this bit will reset the next read value to 0. Writing a 0 to this bit has no effect. Software can poll this bit until it reads a 0, and will then own the usage of the host controller.				RW
05	SMBALERT_STS	0 = Interrupt or SMI# was not generated by SMBALERT#. Software clears this bit by writing a 1 to it. 1 = The source of the interrupt or SMI# was the SMBALERT# signal. This bit is only cleared by software writing a 1 to the bit position or by RSMRST# going low. If the signal is programmed as a GPIO, then this bit will never be set.				RWC
04	FAILED	0 = Software clears this bit by writing a 1 to it. 1 = The source of the interrupt or SMI# was a failed bus transaction. This bit is set in response to the KILL bit being set to terminate the host transaction.				RWC
03	BUS_ERR	0 = Software clears this bit by writing a 1 to it. 1 = The source of the interrupt of SMI# was a transaction collision.				RWC

Table 11-19. Offset 00h: Host Status Register (SMBus—B0:D31:F3) (Sheet 2 of 2)

Description:						
View: PCI		**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit		**Default:** 00h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	02	DEV_ERR	0 = Software clears this bit by writing a 1 to it. The PCH will then deassert the interrupt or SMI#. 1 = The source of the interrupt or SMI# was due to one of the following: Invalid Command Field, Unclaimed Cycle (host initiated), Host Device Time-out Error.			RWC
	01	INTR	This bit can only be set by termination of a command. INTR is not dependent on the INTREN bit (offset SMBASE + 02h, bit 0) of the Host controller register (offset 02h). It is only dependent on the termination of the command. If the INTREN bit is not set, then the INTR bit will be set, although the interrupt will not be generated. Software can poll the INTR bit in this non-interrupt case. 0 = Software clears this bit by writing a 1 to it. The PCH then deasserts the interrupt or SMI#. 1 = The source of the interrupt or SMI# was the successful completion of its last command.			RWC
	00	HOST_BUSY	0 = Cleared by the PCH when the current transaction is completed. 1 = Indicates that the PCH is running a command from the host interface. No SMB registers should be accessed while this bit is set, except the BLOCK DATA BYTE Register. The BLOCK DATA BYTE Register can be accessed when this bit is set only when the SMB_CMD bits in the Host Control Register are programmed for Block command or I^2C Read command. This is necessary in order to check the DONE_STS bit.			RWC

11.2.1.2 Offset 02h: Host Control Register (SMBus—B0:D31:F3)

A read to this register will clear the byte pointer of the 32-byte buffer.

Table 11-20. Offset 02h: Host Control Register (SMBus—B0:D31:F3) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	PEC_EN	0 = SMBus host controller does not perform the transaction with the PEC phase appended. 1 = Causes the host controller to perform the SMBus transaction with the Packet Error Checking phase appended. For writes, the value of the PEC byte is transferred from the PEC Register. For reads, the PEC byte is loaded in to the PEC Register. This bit must be written prior to the write in which the START bit is set.				RW
06	START	0 = This bit will always return 0 on reads. The HOST_BUSY bit in the Host Status register (offset 00h) can be used to identify when the PCH has finished the command. 1 = Writing a 1 to this bit initiates the command described in the SMB_CMD field. All registers should be setup prior to writing a 1 to this bit position.				WO
05	LAST_BYTE	This bit is used for Block Read commands. 1 = Software sets this bit to indicate that the next byte will be the last byte to be received for the block. This causes the PCH to send a NACK (instead of an ACK) after receiving the last byte. Once the SECOND_TO_STS bit in TCO2_STS register (B0:D31:F0, TCOBASE+6h, bit 1) is set, the LAST_BYTE bit also gets set. While the SECOND_TO_STS bit is set, the LAST_BYTE bit cannot be cleared. This prevents the PCH from running some of the SMBus commands (Block Read/Write, I^2C Read, Block I^2C Write).				WC

Table 11-20. Offset 02h: Host Control Register (SMBus—B0:D31:F3) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
04:02	SMB_CMD	The bit encoding below indicates which command the PCH is to perform. If enabled, the PCH will generate an interrupt or SMI# when the command has completed If the value is for a non-supported or reserved command, the PCH will set the device error (DEV_ERR) status bit (offset SMBASE + 00h, bit 2) and generate an interrupt when the START bit is set. The PCH will perform no command, and will not operate until DEV_ERR is cleared. 000 = **Quick**: The slave address and read/write value (bit 0) are stored in the transmit slave address register. 001 = **Byte**: This command uses the transmit slave address and command registers. Bit 0 of the slave address register determines if this is a read or write command. 010 = **Byte Data**: This command uses the transmit slave address, command, and DATA0 registers. Bit 0 of the slave address register determines if this is a read or write command. If it is a read, the DATA0 register will contain the read data. 011 = **Word Data**: This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. If it is a read, after the command completes, the DATA0 and DATA1 registers will contain the read data. 100 = **Process Call:** This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. After the command completes, the DATA0 and DATA1 registers will contain the read data. 101 = **Block**: This command uses the transmit slave address, command, DATA0 registers, and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block reads, the count is received and stored in the DATA0 register. Bit 0 of the slave address register selects if this is a read or write command. For writes, data is retrieved from the first n (where n is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. 110 = **I^2C Read**: This command uses the transmit slave address, command, DATA0, DATA1 registers, and the Block Data Byte register. The read data is stored in the Block Data Byte register. The PCH continues reading data until the NAK is received. 111 **= Block Process:** This command uses the transmit slave address, command, DATA0 and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block read, the count is received and stored in the DATA0 register. Bit 0 of the slave address register always indicate a write command. For writes, data is retrieved from the first m (where m is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. E32B bit in the Auxiliary Control register must be set for this command to work.			RW
01	KILL	0 = Normal SMBus host controller functionality. 1 = Kills the current host transaction taking place, sets the FAILED status bit, and asserts the interrupt (or SMI#). This bit, once set, must be cleared by software to allow the SMBus host controller to function normally.				RW
00	INTREN	0 = Disable. 1 = Enable the generation of an interrupt or SMI# upon the completion of the command.				RW

11.2.1.3 Offset 03h: Host Command Register (SMBus—B0:D31:F3)

Table 11-21. Offset 03h: Host Command Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 03h **Offset End:** 03h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		This 8-bit field is transmitted by the host controller in the command field of the SMBus protocol during the execution of any command.			RW

11.2.1.4 Offset 04h: Transmit Slave Address Register (SMBus—B0:D31:F3)

This register is transmitted by the host controller in the slave address field of the SMBus protocol.

Table 11-22. Offset 04h: Transmit Slave Address Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 04h **Offset End:** 04h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	ADDRESS	This field provides a 7-bit address of the targeted slave.			RW
00	RW	Direction of the host transfer. 0 = Write 1 = Read			RW

11.2.1.5 Offset 05h: Host Data 0 Register (SMBus—B0:D31:F3)

Table 11-23. Offset 05h: Host Data 0 Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 05h **Offset End:** 05h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Data0/Count	This field contains the 8-bit data sent in the DATA0 field of the SMBus protocol. For block write commands, this register reflects the number of bytes to transfer. This register should be programmed to a value between 1 and 32 for block counts. A count of 0 or a count above 32 will result in unpredictable behavior. The host controller does not check or log invalid block counts.			RW

11.2.1.6 Offset 06h: Host Data 1 Register (SMBus—B0:D31:F3)

Table 11-24. Offset 06h: Host Data 1 Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 06h **Offset End:** 06h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	Data1	This 8-bit register is transmitted in the DATA1 field of the SMBus protocol during the execution of any command.			RW

11.2.1.7 Offset 07h: Host Block Data Byte Register (SMBus—B0:D31:F3)

Table 11-25. Offset 07h: Host Block Data Byte Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)	**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 07h **Offset End:** 07h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BDTA	This is either a register, or a pointer into a 32-byte block array, depending upon whether the E32B bit is set in the Auxiliary Control register. When the E32B bit (offset SMBASE + 0Dh, bit 1) is cleared, this is a register containing a byte of data to be sent on a block write or read from on a block read, just as it behaved on the ICH3. When the E32B bit is set, reads and writes to this register are used to access the 32-byte block data storage array. An internal index pointer is used to address the array, which is reset to 0 by reading the HCTL register (offset 02h). The index pointer then increments automatically upon each access to this register. The transfer of block data into (read) or out of (write) this storage array during an SMBus transaction always starts at index address 0. When the E2B bit is set, for writes, software will write up to 32-bytes to this register as part of the setup for the command. After the Host controller has sent the Address, Command, and Byte Count fields, it will send the bytes in the SRAM pointed to by this register. When the E2B bit is cleared for writes, software will place a single byte in this register. After the host controller has sent the address, command, and byte count fields, it will send the byte in this register. If there is more data to send, software will write the next series of bytes to the SRAM pointed to by this register and clear the DONE_STS bit. The controller will then send the next byte. During the time between the last byte being transmitted to the next byte being transmitted, the controller will insert wait-states on the interface. When the E2B bit is set for reads, after receiving the byte count into the Data0 register, the first series of data bytes go into the SRAM pointed to by this register. If the byte count has been exhausted or the 32-byte SRAM has been filled, the controller will generate an SMI# or interrupt (depending on configuration) and set the DONE_STS bit. Software will then read the data. During the time between when the last byte is read from the SRAM to when the DONE_STS bit is cleared, the controller will insert wait-states on the interface.			RW

11.2.1.8 Offset 08h: Packet Error Check (PEC) Register (SMBus—B0:D31:F3)

Table 11-26. Offset 08h: Packet Error Check (PEC) Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PEC_DATA	This 8-bit register is written with the 8-bit CRC value that is used as the SMBus PEC data prior to a write transaction. For read transactions, the PEC data is loaded from the SMBus into this register and is then read by software. Software must ensure that the INUSE_STS bit is properly maintained to avoid having this field over-written by a write transaction following a read transaction.			RW

11.2.1.9 Offset 09h: Receive Slave Address Register (SMBus—B0:D31:F3)

Table 11-27. Offset 09h: Receive Slave Address Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 44h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	Reserved	Reserved			RW
06 :00	SLAVE_ADDR	This field is the slave address that the PCH decodes for read and write cycles. the default is not 0, so the SMBus Slave Interface can respond even before the processor comes up (or if the processor is dead). This register is cleared by RSMRST#, but not by PLTRST#.			RW

11.2.1.10 Offset 0Ah: Receive Slave Data Register (SMBus—B0:D31:F3)

This register contains the 16-bit data value written by the external SMBus master. The processor can then read the value from this register. This register is reset by RSMRST#, but not PLTRST#

Table 11-28. Offset 0Ah: Receive Slave Data Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)	**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Ah **Offset End:** 0Bh		
Size: 8 bit	**Default:** 0000h		**Power Well:** Resume		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	DATA_MSG1	Data Message Byte 1			RO
07 :00	DATA_MSG0	Data Message Byte 0			RO

11.2.1.11 Offset 0Ch: Auxiliary Status Register (SMBus—B0:D31:F3)

Table 11-29. Offset 0Ch: Auxiliary Status Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)	**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Ch **Offset End:** 0Ch		
Size: 8 bit	**Default:** 00h		**Power Well:** Resume		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :02	Reserved	Reserved			
01	STCO	SMBus TCO Mode — This bit reflects the strap setting of TCO compatible mode vs. Advanced TCO mode. 0 = The PCH is in the compatible TCO mode. 1 = Reserved			RO
00	CRCE	CRC Error: 0 = Software clears this bit by writing a 1 to it. 1 = This bit is set if a received message contained a CRC error. When this bit is set, the DERR bit of the host status register will also be set. This bit will be set by the controller if a software abort occurs in the middle of the CRC portion of the cycle or an abort happens after the PCH has received the final data bit transmitted by an external slave.			RWC

11.2.1.12 Offset 0Dh: Auxiliary Control Register (SMBus—B0:D31:F3)

Table 11-30. Offset 0Dh: Auxiliary Control Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :02	Reserved	Reserved			
01	E32B	Enable 32-Byte Buffer: 0 = Disable. 1 = Enable. When set, the Host Block Data register is a pointer into a 32-byte buffer, as opposed to a single register. This enables the block commands to transfer or receive up to 32-bytes before the PCH generates an interrupt.			RW
00	AAC	Automatically Append CRC: 0 = The PCH will Not automatically append the CRC. 1 = The PCH will automatically append the CRC. This bit must not be changed during SMBus transactions or undetermined behavior will result. It should be programmed only once during the lifetime of the function.			RW

11.2.1.13 Offset 0Eh: SMLink Pin Control Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#. This register is only applicable in the TCO compatible mode.

Table 11-31. Offset 0Eh: SMLink Pin Control Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)	**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** See register description			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	**Reserved**			
02	SMLINK_CLK_CTL	0 = The PCH will drive the SMLINK0 pin low, independent of what the other SMLink logic would otherwise indicate for the SMLINK0 pin. 1 = The SMLINK0 pin is **not** overdriven low. The other SMLink logic controls the state of the pin. (Default)			RW
01	SMLINK1_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMLINK1 pin. This allows software to read the current state of the pin. 0 = Low 1 = High			RO
00	SMLINK0_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMLINK0 pin. This allows software to read the current state of the pin. 0 = Low 1 = High			RO

11.2.1.14 Offset 0Fh: SMBus Pin Control Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

Table 11-32. Offset 0Fh: SMBus Pin Control Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 0Fh **Offset End:** 0Fh	
Size: 8 bit	**Default:** See register description			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	**Reserved**			
02	SMBCLK_CTL	1 = The SMBCLK pin is **not** overdriven low. The other SMBus logic controls the state of the pin. 0 = The PCH drives the SMBCLK pin low, independent of what the other SMB logic would otherwise indicate for the SMBCLK pin. (Default)			RW
01	SMBDATA_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMBDATA pin. This allows software to read the current state of the pin. 0 = Low 1 = High			RO
00	SMBCLK_CUR_STS	This read-only bit has a default value that is dependent on an external signal level. This pin returns the value on the SMBCLK pin. This allows software to read the current state of the pin. 0 = Low 1 = High			RO

11.2.1.15 Offset 10h: Slave Status Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

All bits in this register are implemented in the 64 kHz clock domain. Therefore, software must poll this register until a write takes effect before assuming that a write has completed internally.

Table 11-33. Offset 10h: Slave Status Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 10h **Offset End:** 10h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved			
00	HOST_NOTIFY_STS	The PCH sets this bit to a 1 when it has completely received a successful Host Notify Command on the SMBus pins. Software reads this bit to determine that the source of the interrupt or SMI# was the reception of the Host Notify Command. Software clears this bit after reading any information needed from the Notify address and data registers by writing a 1 to this bit. The PCH will allow the Notify Address and Data registers to be over-written once this bit has been cleared. When this bit is 1, the PCH will NACK the first byte (host address) of any new “Host Notify” commands on the SMBus pins. Writing a 0 to this bit has no effect.			RWC

11.2.1.16 Offset 11h: Slave Command Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

Table 11-34. Offset 11h: Slave Command Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)	**Bus:Device:Function:** B0:D31:F3		**Offset Start:** 11h **Offset End:** 11h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved			
02	SMBALERT_DIS	0 = Allows the generation of the interrupt or SMI#. 1 = Software sets this bit to block the generation of the interrupt or SMI# due to the SMBALERT# source. This bit is logically inverted and ANDed with the SMBALERT_STS bit (offset SMBASE + 00h, bit 5). The resulting signal is distributed to the SMI# and/or interrupt generation logic. This bit does not effect the wake logic.			RW
01	HOST_NOTIFY_WKEN	Software sets this bit to 1 to enable the reception of a Host Notify command as a wake event. When enabled this event is "OR'd" in with the other SMBus wake events and is reflected in the SMB_WAK_STS bit of the General Purpose Event 0 Status register. 0 = Disable 1 = Enable			RW
00	HOST_NOTIFY_INTREN	Software sets this bit to 1 to enable the generation of interrupt or SMI# when HOST_NOTIFY_STS (offset SMBASE + 10h, bit 0) is 1. This enable does not affect the setting of the HOST_NOTIFY_STS bit. When the interrupt is generated, either PIRQB# or SMI# is generated, depending on the value of the SMB_SMI_EN bit (B0:D31:F3:40h, bit 1). If the HOST_NOTIFY_STS bit is set when this bit is written to a 1, then the interrupt (or SMI#) will be generated. The interrupt (or SMI#) is logically generated by AND'ing the STS and INTREN bits. 0 = Disable 1 = Enable			RW

11.2.1.17 Offset 14h: Notify Device Address Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

Table 11-35. Offset 14h: Notify Device Address Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 14h **Offset End:** 14h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	DEVICE_ADDRESS	This field contains the 7-bit device address received during the Host Notify protocol of the SMBus 2.0 Specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit (B0:D31:F3:SMBASE +10, bit 0) is set to 1.			RO
00	Reserved	Reserved			

11.2.1.18 Offset 16h: Notify Data Low Byte Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

Table 11-36. Offset 16h: Notify Data Low Byte Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 16h **Offset End:** 16h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	DATA_LOW_BYTE	This field contains the first (low) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit (B0:D31:F3:SMBASE +10, bit 0) is set to 1.			RO

11.2.1.19 Offset 17h: Notify Data High Byte Register (SMBus—B0:D31:F3)

This register is in the resume well and is reset by RSMRST#.

Table 11-37. Offset 17h: Notify Data High Byte Register (SMBus—B0:D31:F3)

Description:					
View: PCI	**BAR:** SMBASE (IO)		**Bus:Device:Function:** B0:D31:F3	**Offset Start:** 17h **Offset End:** 17h	
Size: 8 bit	**Default:** 00h			**Power Well:** Resume	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	DATA_HIGH_BYTE	This field contains the second (high) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit (B0:D31:F3:SMBASE +10, bit 0) is set to 1.			RO

§ §

12.0 PCI Express* Configuration Registers (B0:D28:F0/1/2/3)

12.1 PCI Express* Configuration Registers (PCI Express* — B0:D28:F0/F1/F2/F3)

Note: Register address locations that are not shown in the following table should be treated as Reserved.

12.1.1 PCI Express* Configuration Registers Address Map

Table 12-1. PCI Express* Configuration Registers Address Map (Sheet 1 of 4)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 627	8086h
02h	03h	"Offset 02h: DID—Device Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 627	See Table 3-3
04h	05h	"Offset 04h: PCI COMMAND Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 628	0000h
06h	07h	"Offset 06h: PCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 629	0010h
08h	08h	"Offset 08h: RID—Revision Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 631	See register description
09h	09h	"Offset 09h: Programming Interface Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 631	00h
0Ah	0Ah	"Offset 0Ah: Sub Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 632	04h
0Bh	0Bh	"Offset 0Bh: Base Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 632	06h
0Ch	0Ch	"Offset 0Ch: Cache Line Size Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 633	00h
0Dh	0Dh	"Offset 0Dh: Primary Latency Timer Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 633	00h
0Eh	0Eh	"Offset 0Eh: Header Type Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 634	81h
18h	1Ah	"Offset 18h: Bus Number Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 634	000000h
1Bh	1Bh	"Offset 1Bh: Secondary Latency Timer (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 635	00h
1Ch	1Dh	"Offset 1Ch: IOBL—I/O Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 635	0000h
1Eh	1Fh	"Offset 1Eh: Secondary Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 636	0000h

Table 12-1. PCI Express* Configuration Registers Address Map (Sheet 2 of 4)

Offset Start	Offset End	Register ID - Description	Default Value
20h	23h	"Offset 20h: Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 637	00000000h
24h	27h	"Offset 24h: Prefetchable Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 638	00010001h
28h	2Bh	"Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 638	00000000h
2Ch	2Fh	"Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 639	00000000h
34h	40h	"Offset 34h: Capabilities List Pointer Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 639	8086h
3Ch	3Dh	"Offset 3Ch: Interrupt Information Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 640	See register description
3Eh	3fh	"Offset 3Eh: Bridge Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 640	0000h
40h	41h	"Offset 40h: Capabilities List Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 642	8010h
42h	43h	"Offset 42h: PCI Express* Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 642	0042h
44h	47h	"Offset 44h: Device Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 643	00008000h
48h	49h	"Offset 48h: Device Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 644	0000h
4Ah	4Bh	"Offset 4Ah: Device Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 645	0010h
4Ch	4Fh	"Offset 4Ch: Link Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 646	See register description
50h	51h	"Offset 50h: Link Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 648	0000h
52h	53h	"Offset 52h: Link Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 649	See register description
54h	57h	"Offset 54h: Slot Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 650	00040060h
58h	59h	"Offset 58h: Slot Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 651	0000h
5Ah	5Bh	"Offset 5Ah: Slot Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 652	0000h
5Ch	5Dh	"Offset 5Ch: Root Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 653	0000h
60h	63h	"Offset 60h: Root Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 654	00000000h
64h	67h	"Offset 64h: Device Capabilities 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 654	00000016h
68h	69h	"Offset 68h: Device Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 655	0000h
70h	71h	"Offset 70h: Link Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 656	0001h
80h	81h	"Offset 80h: Message Signaled Interrupt Identifiers Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 656	9005h
82h	83h	"Offset 82h: Message Signaled Interrupt Message Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 657	0000h

Table 12-1. PCI Express* Configuration Registers Address Map (Sheet 3 of 4)

Offset Start	Offset End	Register ID - Description	Default Value
84h	87h	"Offset 84h: Message Signaled Interrupt Message Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 658	00000000h
88h	89h	"Offset 88h: Message Signaled Interrupt Message Data Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 658	0000h
90h	91h	"Offset 90h: Subsystem Vendor Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 659	A00Dh
94h	97h	"Offset 94h: Subsystem Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 659	00000000h
A0h	A1h	"Offset A0h: Power Management Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 660	0001h
A2h	A3h	"Offset A2h: PCI Power Management Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 660	C802h
A4h	A7h	"Offset A4h: PCI Power Management Control and Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 661	00000000h
D4h	D7h	"Offset D4h: Miscellaneous Port Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 662	00000000h
D8h	DBh	"Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 663	08110000h
DCh	DFh	"Offset DCh: SMI/SCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 666	00000000h
E1h	E1h	"Offset E1h: Root Port Dynamic Clock Gating Enable (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 667	00h
E8h	EBh	"Offset E8h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 668	00000020h
104h	107h	"Offset 104h: Uncorrectable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 668	00000000000 x0xxx0x0x00 00000x0000b
108h	10Bh	"Offset 108h: Uncorrectable Error Mask (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 670	00000000h
10Ch	10Fh	"Offset 10Ch: Uncorrectable Error Severity (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 672	00060011h
110h	113h	"Offset 110h: Correctable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 673	00000000h
114h	117h	"Offset 114h: Correctable Error Mask Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 674	00002000h
118h	11Bh	"Offset 118h: Advanced Error Capabilities and Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 675	00000000h
130h	133h	"Offset 130h: Root Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 676	00000000h
180h	183h	"Offset 180h: Root Complex Topology Capability List Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 677	00010005h
184h	187h	"Offset 184h: Element Self Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 677	See register description
190h	193h	"Offset 190h: Upstream Link Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 678	8086h
198h	19Fh	"Offset 198h: Upstream Link Base Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 679	See register description

Table 12-1. PCI Express* Configuration Registers Address Map (Sheet 4 of 4)

Offset Start	Offset End	Register ID - Description	Default Value
300h	303h	"Offset 300h: PCI Express* Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 679	60005007h
318h	318h	"Offset 318h: PCI Express* Extended Test Mode Register (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 680	See register description
324h	324h	"OFfset 324h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)" on page 680	14000016h

12.1.1.1 Offset 00h: Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-2. Offset 00h: Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

12.1.1.2 Offset 02h: DID—Device Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-3. Offset 02h: DID—Device Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** See Table 3-3			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device Identification: This is a 16-bit value assigned to the PCH LPC bridge controller.		See Table 3-3	RO

12.1.1.3 Offset 04h: PCI COMMAND Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-4. Offset 04h: PCI COMMAND Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved			
10	ID	Interrupt Disable: This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management ***and*** MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set			RW
09	FBE	Fast Back to Back Enable — Reserved per the *PCI Express* Base Specification*.			RO
08	SEE	SERR# Enable: When set, enables the root port to generate an SERR# message when PSTS.SSE is set.			RW
07	WCC	Wait Cycle Control — Reserved per the *PCI Express* Base Specification*.			RO
06	PERE	Parity Error Response Enable: Indicates that the device is capable of reporting parity errors as a master on the backbone			RW
05	VPS	VGA Palette Snoop— Reserved per the *PCI Express* Base Specification*.			RO
04	MWIE	Memory Write Invalidate Enable — Reserved per the *PCI Express* Base Specification*.			RO
03	SCE	Special Cycle Enable — Reserved per the *PCI Express* Base Specification*.			RO

Table 12-4. Offset 04h: PCI COMMAND Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BME	Bus Master Enable: When set, allows the root port to forward cycles onto the backbone from a PCI-Express device. When cleared, all cycles from the device are master aborted			RW
01	MSE	Memory Space Enable: When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are master aborted on the backbone			RW
00	IOSE	I/O Space Enable — When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are master aborted the backbone.		0	RW

12.1.1.4 Offset 06h: PCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-5. Offset 06h: PCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = No parity error detected. 1 = Set when the root port receives a command or data from the backbone with a parity error. This is set even if PCIMD.PER (B0:D28:F0/F1/F2/F3:04, bit 6) is not set.			RWC
14	SSE	Signaled System Error: 0 = No system error signaled. 1 = Set when the root port signals a system error to the internal SERR# logic.			RWC
13	RMA	Received Master Abort: 0 = Root port has not received a completion with unsupported request status from the backbone. 1 = Set when the root port receives a completion with unsupported request status from the backbone.			RWC

Table 12-5. Offset 06h: PCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
12	RTA	Received Target Abort: 0 = Root port has not received a completion with completer abort from the backbone. 1 = Set when the root port receives a completion with completer abort from the backbone.			RWC
11	STA	Signaled Target Abort: 0 = No target abort received. 1 = Set whenever the root port forwards a target abort received from the downstream device onto the backbone.			RWC
10 :09	DPED	Master Data Parity Error Detected: 0 = No data parity error received. 1 = Set when the root port receives a completion with a data parity error on the backbone and PCIMD.PER (B0:D28:F0/F1/F2/F3:04, bit 6) is set.			RWC
08	FB2BC	Fast Back to Back Capable — Reserved per the *PCI Express* Base Specification*.			
07	Reserved	Reserved			
06	SC66	66 MHz Capable — Reserved per the *PCI Express* Base Specification*.			
05		Capabilities List — Hardwired to 1. Indicates the presence of a capabilities list.			RO
04		Interrupt Status — Indicates status of Hot-Plug and power management interrupts on the root port that result in INTx# message generation. 0 = Interrupt is deasserted. 1 = Interrupt is asserted. This bit is not set if MSI is enabled. If MSI is not enabled, this bit is set regardless of the state of PCICMD.Interrupt Disable bit (B0:D28:F0/F1/F2/F3:04h:bit 10).			RO
03	Reserved	Reserved			
02 :00	DPE	Detected Parity Error:. 0 = No parity error detected. 1 = Set when the root port receives a command or data from the backbone with a parity error. This is set even if PCIMD.PER (B0:D28:F0/F1/F2/F3:04, bit 6) is not set.			RWC

12.1.1.5 Offset 08h: RID—Revision Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-6. Offset 08h: RID—Revision Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: This is an 8-bit value indicating the stepping of PCIe* controller hardware.			RO

12.1.1.6 Offset 09h: Programming Interface Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-7. Offset 09h: Programming Interface Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PI	Programming Interface: 00h = No specific register level programming interface defined.			RO

12.1.1.7 Offset 0Ah: Sub Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-8. Offset 0Ah: Sub Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 04h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code — This field is determined by bit 2 of the MPC register (B0:D28:F0-5:Offset D8h, bit 2). 04h = PCI-to-PCI bridge. 00h = Host Bridge.			RO

12.1.1.8 Offset 0Bh: Base Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-9. Offset 0Bh: Base Class Code Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 06h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code — 8-bit value that indicates the type of device for the LPC bridge. 06h = Bridge device.			RO

12.1.1.9 Offset 0Ch: Cache Line Size Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-10. Offset 0Ch: Cache Line Size Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CLS	Cache Line Size — This is read/write but contains no functionality, per the *PCI Express* Base Specification*.			RW

12.1.1.10 Offset 0Dh: Primary Latency Timer Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-11. Offset 0Dh: Primary Latency Timer Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03		Latency Count. Reserved per the *PCI Express* Base Specification.*			RO
02 :00	Reserved	Reserved			

12.1.1.11 Offset 0Eh: Header Type Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-12. Offset 0Eh: Header Type Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 81h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Multi-Function Device: 0 = Single-function device. 1 = Multi-function device.			RO
06 :00		Configuration Layout — This field is determined by bit 2 of the MPC register (B0:D28:F0-5:Offset D8h, bit 2). 00h = Indicates a Host Bridge. 01h = Indicates a PCI-to-PCI bridge.			RO

12.1.1.12 Offset 18h: Bus Number Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-13. Offset 18h: Bus Number Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 18h **Offset End:** 1Ah	
Size: 24 bit	**Default:** 000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16	SBBN	Subordinate Bus Number — Indicates the highest PCI bus number below the bridge.			RW
15 :08	SCBN	Secondary Bus Number — Indicates the bus number the port.			RW
07 :00	PBN	Primary Bus Number — Indicates the bus number of the backbone.			RW

12.1.1.13 Offset 1Bh: Secondary Latency Timer (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-14. Offset 1Bh: Secondary Latency Timer (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 1Bh **Offset End:** 1Bh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00		Secondary Latency Timer — Reserved for a Root Port per the *PCI Express* Base Specification*			RO

12.1.1.14 Offset 1Ch: IOBL—I/O Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-15. Offset 1Ch: IOBL—I/O Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 1Ch **Offset End:** 1Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	IOLA	I/O Limit Address — I/O Base bits corresponding to address lines 15:12 for 4-KB alignment. Bits 11:0 are assumed to be padded to FFFh.			RW
11 :08	IOLC	I/O Limit Address Capability — Indicates that the bridge does not support 32-bit I/O addressing.			RO
07 :04	IOBA	I/O Base Address — I/O Base bits corresponding to address lines 15:12 for 4-KB alignment. Bits 11:0 are assumed to be padded to 000h.			RW
03 :00	IOBC	I/O Base Address Capability — Indicates that the bridge does not support 32-bit I/O addressing.			RO

12.1.1.15 Offset 1Eh: Secondary Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-16. Offset 1Eh: Secondary Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 1Eh **Offset End:** 1Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = No error. 1 = The port received a poisoned TLP.			RWC
14	RSE	Received System Error: 0 = No error. 1 = The port received an ERR_FATAL or ERR_NONFATAL message from the device.			RWC
13	RMA	Received Master Abort: 0 = Unsupported Request not received. 1 = The port received a completion with "Unsupported Request" status from the device.			RWC
12	RTA	Received Target Abort: 0 = Completion Abort not received. 1 = The port received a completion with "Completion Abort" status from the device.			RWC
11	STA	Signaled Target Abort: 0 = Completion Abort not sent. 1 = The port generated a completion with "Completion Abort" status to the device.			RWC
10 :09	SDTS	Secondary DEVSEL# Timing Status: Reserved per *PCI Express* Base Specification*.			
08	DPD	Data Parity Error Detected: 0 = Conditions below did not occur. 1 = Set when the BCTRL.PERE (B0:D28:FO/F1/F2/F3:3E: bit 0) is set, and either of the following two conditions occurs: -Port receives completion marked poisoned. -Port poisons a write request to the secondary side.			RWC
07	SFBC	Secondary Fast Back to Back Capable: Reserved per *PCI Express* Base Specification*.			
16	Reserved	Reserved			
05	SC66	Secondary 66 MHz Capable: Reserved per *PCI Express* Base Specification*.			
04 :00	Reserved	Reserved			

12.1.1.16 Offset 20h: Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE (B0:D28:F0/F1/F2/F3:04:bit 1) is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME (B0:D28:F0/F1/F2/F3:04:bit 2) is set. The comparison performed is MB ≥ AD[31:20] ≤ ML.

Table 12-17. Offset 20h: Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ML	Memory Limit — These bits are compared with bits 31:20 of the incoming address to determine the upper 1-MB aligned value of the range.			RW
19 16	Reserved	Reserved			
15 :04	MB	Memory Base — These bits are compared with bits 31:20 of the incoming address to determine the lower 1-MB aligned value of the range.			RW
03 :00	Reserved	Reserved			

12.1.1.17 Offset 24h: Prefetchable Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE (B0:D28:F0/F1/F2/F3;04, bit 1) is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME (B0:D28:F0/F1/F2/F3;04, bit 2) is set. The comparison performed is PMBU32:PMB ≥ AD[63:32]:AD[31:20] ≤ PMLU32:PML.

Table 12-18. Offset 24h: Prefetchable Memory Base and Limit Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00010001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	PML	Prefetchable Memory Limit — These bits are compared with bits 31:20 of the incoming address to determine the upper 1-MB aligned value of the range.			RW
19 16	I64L	64-bit Indicator — Indicates support for 64-bit addressing			RO
15 :04	PMB	Prefetchable Memory Base — These bits are compared with bits 31:20 of the incoming address to determine the lower 1-MB aligned value of the range.			RW
03 :00	I64B	64-bit Indicator — Indicates support for 64-bit addressing			RO

12.1.1.18 Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-19. Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 28h **Offset End:** 2Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PMBU	Prefetchable Memory Base Upper Portion — Upper 32-bits of the prefetchable address base.			RW

12.1.1.19 Offset 2Ch: Prefetchable Memory Limit Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-20. Offset 28h: PMBU32—Prefetchable Memory Base Upper 32 Bits Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 2Ch **Offset End:** 2Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PMBU	Prefetchable Memory Base Upper Portion — Upper 32-bits of the prefetchable address base.			RW

12.1.1.20 Offset 34h: Capabilities List Pointer Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-21. Offset 34h: Capabilities List Pointer Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 34h **Offset End:** 40h	
Size: 8 bit	**Default:** 8086h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PTR	Capabilities Pointer — Indicates that the pointer for the first entry in the capabilities list is at 40h in configuration space.			RO

12.1.1.21 Offset 3Ch: Interrupt Information Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-22. Offset 3Ch: Interrupt Information Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 3Ch **Offset End:** 3Dh	
Size: 16 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	IPIN	Interrupt Pin — Indicates the interrupt pin driven by the root port. At reset, this register takes on the following values, which reflect the reset state of the B0:D28IP register in config space: Port: Reset Value 1: D28IP.P1IP 2: D28IP.P2IP 3: D28IP.P3IP 4: D28IP.P4IP The value that is programmed into B0:D28IP is always reflected in this register.			RO
07 :00	ILINE	Interrupt Line — Default = 00h. Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register. These bits are not reset by FLR.			RW

12.1.1.22 Offset 3Eh: Bridge Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-23. Offset 3Eh: Bridge Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 3Eh **Offset End:** 3fh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved			
11	DTSE	Discard Timer SERR# Enable: Reserved per *PCI Express* Base Specification,* Revision 1.0a			
10	DTS	Discard Timer Status: Reserved per *PCI Express* Base Specification,* Revision 1.0a.			

Table 12-23. Offset 3Eh: Bridge Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 3Eh **Offset End:** 3fh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	SDT	Secondary Discard Timer: Reserved per *PCI Express* Base Specification,* Revision 1.0a.			
08	PDT	Primary Discard Timer — Reserved per *PCI Express* Base Specification,* Revision 1.0a.			
07	FBE	Fast Back to Back Enable — Reserved per *PCI Express* Base Specification,* Revision 1.0a.			
06	SBR	Secondary Bus Reset — Triggers a hot reset on the PCI Express* port.			RW
05	MAM	Master Abort Mode — Reserved per Express specification.			
04	V16	VGA 16-Bit Decode: 0 = VGA range is enabled. 1 = The I/O aliases of the VGA range (See BCTRL:VE definition below), are not enabled, and only the base I/O ranges can be decoded			RW
03	VE	VGA Enable: 0 = The ranges below will not be claimed off the backbone by the root port. 1 = The following ranges will be claimed off the backbone by the root port: -Memory ranges A0000h-BFFFFh -I/O ranges 3B0h – 3BBh and 3C0h – 3DFh, and all aliases of bits 15:10 in any combination of 1s			RW
02	IE	ISA Enable — This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64 KB of PCI I/O space. 0 = The root port will not block any forwarding from the backbone as described below. 1 = The root port will block any forwarding from the backbone to the device of I/O transactions addressing the last 768 bytes in each 1-KB block (offsets 100h to 3FFh).			RW
01	SE	SERR# Enable: 0 = The messages described below are not forwarded to the backbone. 1 = ERR_COR, ERR_NONFATAL, and ERR_FATAL messages received are forwarded to the backbone.			RW
00	PERE	Parity Error Response Enable — When set: 0 = Poisoned write TLPs and completions indicating poisoned TLPs will not set the SSTS.DPD (B0:D28:F0/F1/F2/F3:1E, bit 8). 1 = Poisoned write TLPs and completions indicating poisoned TLPs will set the SSTS.DPD (B0:D28:F0/F1/F2/F3:1E, bit 8).			RW

12.1.1.23 Offset 40h: Capabilities List Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-24. Offset 40h: Capabilities List Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 40h **Offset End:** 41h	
Size: 16 bit	**Default:** 8010h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — Value of 80h indicates the location of the next pointer.			RO
07 :00	CID	Capability ID — Indicates this is a PCI Express* capability.			RO

12.1.1.24 Offset 42h: PCI Express* Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-25. Offset 42h: PCI Express* Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 42h **Offset End:** 43h	
Size: 16 bit	**Default:** 0042h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13 :09	IMN	Interrupt Message Number — The PCH does not have multiple MSI interrupt numbers.			RO
08	SI	Slot Implemented — Indicates whether the root port is connected to a slot. Slot support is platform specific. BIOS programs this field, and it is maintained until a platform reset.			RWO
07 :04	DT	Device / Port Type — Indicates this is a PCI Express* root port.			RO
03 :00	CV	Capability Version — Indicates PCI Express* 2.0.			RO

12.1.1.25 Offset 44h: Device Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-26. Offset 44h: Device Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:** 00008000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved			RO
27 :26	CSPS	Captured Slot Power Limit Scale — Not supported.			RO
25 :18	CSPV	Captured Slot Power Limit Value — Not supported.			RO
17 :16	Reserved	Reserved			
15	RBER	Role Based Error Reporting — Indicates that this device implements the functionality defined in the Error Reporting ECN as required by the PCI Express* 1.1 spec.			RO
14 :12	Reserved	Reserved			
11 :09	E1AL	EndPoint L1 Acceptable Latency — This field is reserved with a setting of 000b for devices other than EndPoints, per the PCI Express* 1.1 Spec.			RO
08 :06	E0AL	EndPoint L0s Acceptable Latency — This field is reserved with a setting of 000b for devices other than EndPoints, per the PCI Express* 1.1 Spec.			RO
05	ETFS	Extended Tag Field Supported — Indicates that 8-bit tag fields are supported.			RO
04 :03	PFS	Phantom Functions Supported — No phantom functions supported.			RO
02 :00	MPS	Max Payload Size Supported — Indicates the maximum payload size supported is 128B.			RO

12.1.1.26 Offset 48h: Device Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-27. Offset 48h: Device Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 48h **Offset End:** 49h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved			
14 :12	MRRS	Max Read Request Size — Hardwired to 0.			RO
11	ENS	Enable No Snoop — Not supported. The root port will never issue non-snoop requests.			RO
10	APME	Aux Power PM Enable — The OS will set this bit to 1 if the device connected has detected aux power. It has no effect on the root port otherwise.			RW
09	PFE	Phantom Functions Enable — Not supported.			RO
08	ETFE	Extended Tag Field Enable — Not supported.			RO
07 :05	MPS	Max Payload Size — The root port only supports 128-B payloads, regardless of the programming of this field.			RW
04	ERO	Enable Relaxed Ordering — Not supported.			RO
03	URE	Unsupported Request Reporting Enable: 0 = The root port will ignore unsupported request errors. 1 = Allows signaling ERR_NONFATAL, ERR_FATAL, or ERR_COR to the Root Control register when detecting an unmasked Unsupported Request (UR). An ERR_COR is signaled when a unmasked Advisory Non-Fatal UR is received. An ERR_FATAL, ERR_or NONFATAL, is sent to the Root Control Register when an uncorrectable non-Advisory UR is received with the severity set by the Uncorrectable Error Severity register.			RO
02	FEE	Fatal Error Reporting Enable: 0 = The root port will ignore fatal errors. 1 = Enables signaling of ERR_FATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.			RW
01	NFE	Non-Fatal Error Reporting Enable: 0 = The root port will ignore non-fatal errors. 1 = Enables signaling of ERR_NONFATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.			RW
00	CEE	Correctable Error Reporting Enable: 0 =The root port will ignore correctable errors. 1 = Enables signaling of ERR_CORR to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.			RW

12.1.1.27 Offset 4Ah: Device Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-28. Offset 4Ah: Device Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 4Ah **Offset End:** 4Bh	
Size: 16 bit	**Default:** 0010h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :06	Reserved	Reserved			
05	TDP	Transactions Pending — This bit has no meaning for the root port since only one transaction may be pending to the PCH, so a read of this bit cannot occur until it has already returned to 0.			RO
04	APD	AUX Power Detected — The root port contains AUX power for wakeup.			RO
03	URD	Unsupported Request Detected — Indicates an unsupported request was detected.			RWC
02	FED	Fatal Error Detected — Indicates a fatal error was detected. 0 = Fatal has not occurred. 1 = A fatal error occurred from a data link protocol error, link training error, buffer overflow, or malformed TLP.			RWC
01	NFED	Non-Fatal Error Detected — Indicates a non-fatal error was detected. 0 = Non-fatal has not occurred. 1 = A non-fatal error occurred from a poisoned TLP, unexpected completions, unsupported requests, completer abort, or completer timeout.			RWC
00	CED	Correctable Error Detected — Indicates a correctable error was detected. 0 = Correctable has not occurred. 1 = The port received an internal correctable error from receiver errors / framing errors, TLP CRC error, DLLP CRC error, replay num rollover, replay timeout.			RWC

12.1.1.28 Offset 4Ch: Link Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-29. Offset 4Ch: Link Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 1 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td colspan="2">View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D28 :F0/F1/ F2/F3</td><td>Offset Start: 4Ch
Offset End: 4Fh</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: See register description</td><td></td><td colspan="2">Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :24</td><td>PN</td><td>Port Number — Indicates the port number for the root port. This value is different for each implemented port:
<table><tr><th>Function</th><th>Port #</th><th>Value of PN Field</th></tr><tr><td>D28:F0</td><td>1</td><td>01h</td></tr><tr><td>D28:F1</td><td>2</td><td>02h</td></tr><tr><td>D28:F2</td><td>3</td><td>03h</td></tr><tr><td>D28:F3</td><td>4</td><td>04h</td></tr></table></td><td></td><td></td><td>RO</td></tr>
<tr><td>23 :21</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>20</td><td>LARC</td><td>Link Active Reporting Capable — Hardwired to 1 to indicate that this port supports the optional capability of reporting the DL_Active state of the Data Link Control and Management State Machine.</td><td></td><td></td><td>RO</td></tr>
<tr><td>19 :18</td><td>Reserved</td><td>Reserved</td><td></td><td></td><td></td></tr>
<tr><td>17 :15</td><td>EL1</td><td>L1 Exit Latency — Set to 010b to indicate an exit latency of 2 µs to 4 µs.</td><td></td><td></td><td>RO</td></tr>
<tr><td>14 :12</td><td>EL0</td><td>L0s Exit Latency — Indicates as exit latency based upon common-clock configuration.
<table><tr><th>LCLT.CCC</th><th>Value of EL0 (these bits)</th></tr><tr><td>0</td><td>MPC.UCEL (B0:D28:F0/F1/F2/F3:D8h:bits20:18)</td></tr><tr><td>1</td><td>MPC.CCEL (B0:D28:F0/F1/F2/F3:D8h:bits17:15)</td></tr></table></td><td></td><td></td><td>RO</td></tr>
<tr><td>11 :10</td><td>APMS</td><td>Active State Link PM Support — Indicates what level of active state link power management is supported on the root port.
<table><tr><th>Bits</th><th>Definition</th></tr><tr><td>00b</td><td>Neither L0s nor L1 are supported</td></tr><tr><td>01b</td><td>L0s Entry Supported</td></tr><tr><td>10b</td><td>L1 Entry Supported</td></tr><tr><td>11b</td><td>Both L0s and L1 Entry supported</td></tr></table></td><td></td><td></td><td>RWO</td></tr>
</table>

Table 12-29. Offset 4Ch: Link Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 2 of 2)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR: Configuration</td><td colspan="2">Bus:Device:Function: B0:D28 :F0/F1/ F2/F3</td><td>Offset Start: 4Ch
Offset End: 4Fh</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: See register description</td><td colspan="2"></td><td>Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td>Bit Description</td><td>Sticky</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>09 :04</td><td>MLKW</td><td>Maximum Link Width — For the root ports, several values can be taken, based upon the value of the PCH config register field RPC.PC1 (PCH Config Registers:Offset 0224h:bits1:0) for Ports 1-4 and RPC.PC2 (PCH Config Registers:Offset 0224h:bits1:0) for Ports 5 and 6
<table>
<tr><th colspan="3">Value of MLW Field</th></tr>
<tr><th>Port #</th><th>RPC.PC1=00b</th><th>RPC.PC1=11b</th></tr>
<tr><td>1</td><td>01h</td><td>04h</td></tr>
<tr><td>2</td><td>01h</td><td>01h</td></tr>
<tr><td>3</td><td>01h</td><td>01h</td></tr>
<tr><td>4</td><td>01h</td><td>01h</td></tr>
</table></td><td></td><td></td><td>RO</td></tr>
<tr><td>03 :00</td><td>MLS</td><td>Maximum Link Speed — Set to 1h to indicate the link speed is 2.5 Gb/s.</td><td></td><td></td><td>RO</td></tr>
</table>

12.1.1.29 Offset 50h: Link Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-30. Offset 50h: Link Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 50h **Offset End:** 51h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :10	Reserved	Reserved			
09		Hardware Autonomous Width Disable – Hardware never attempts to change the link width except when attempting to correct unreliable Link operation.			RO
08	Reserved	Reserved			
07	SE	Extended Synch: 0 = Extended synch disabled. 1 = Forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0.			RW
06	CCC	Common Clock Configuration: 0 = The PCH and device are not using a common reference clock. 1 = The PCH and device are operating with a distributed common reference clock.			RW
05	RL	Retrain Link: 0 = This bit always returns 0 when read. 1 = The root port will train its downstream link. Software uses LSTS.LT (B0:D28:F0/F1/F2/F3:52, bit 11) to check the status of training. It is permitted to write 1b to this bit while simultaneously writing modified values to other fields in this register. If the LTSSM is not already in Recovery or Configuration, the resulting Link training must use the modified values. If the LTSSM is already in Recovery or Configuration, the modified values are not required to affect the Link training that is already in progress.			RW
04	LD	Link Disable: 0 = Link enabled. 1 = The root port will disable the link.			RW
03	RCBC	Read Completion Boundary Control) — Indicates the read completion boundary is 64 bytes.			RO
02	Reserved	Reserved			
01 :00	APMC	Active State Link PM Control — Indicates whether the root port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled			RW

12.1.1.30 Offset 52h: Link Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-31. Offset 52h: Link Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved			
13	DLLA	Data Link Layer Active — Default value is 0b. 0 = Data Link Control and Management State Machine is not in the DL_Active state 1 = Data Link Control and Management State Machine is in the DL_Active state			RO
12	SCC	Slot Clock Configuration — Set to 1b to indicate that the PCH uses the same reference clock as on the platform and does not generate its own clock.			RO
11	LT	Link Training — Default value is 0b. 0 = Link training completed. 1 = Link training is occurring.			RO
10	LTE	Link Training Error — Not supported. Set value is 0b.			RO
09 :04	NLW	Negotiated Link Width — This field indicates the negotiated width of the given PCI Express* link. The contents of this NLW field is undefined if the link has not successfully trained. **Port # / Possible Values** 1 — 000001b, 000010b, 000100b 2 — 000001b 3 — 000001b 4 — 000001b 000001b = x1 link width, 000010b =x2 linkwidth, 000100b = x4 linkwidth			RO
03 :00	LS	Link Speed — This field indicates the negotiated Link speed of the given PCI Express* link. 01h = Link is 2.5 Gb/s.			RO

12.1.1.31 Offset 54h: Slot Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-32. Offset 54h: Slot Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 00040060h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	PSN	Physical Slot Number — This is a value that is unique to the slot number. BIOS sets this field and it remains set until a platform reset.			RWO
18 :17	Reserved	Reserved			
16 :15	SLS	Slot Power Limit Scale — Specifies the scale used for the slot power limit value. BIOS sets this field and it remains set until a platform reset.			RWO
14 :07	SLV	Slot Power Limit Value — Specifies the upper limit (in conjunction with SLS value), on the upper limit on power supplied by the slot. The two values together indicate the amount of power in watts allowed for the slot. BIOS sets this field and it remains set until a platform reset.			RWO
06	HPC	Hot Plug Capable: 1b = Indicates that Hot-Plug is supported.			RWO
05	HPS	Hot Plug Surprise: 1b = Indicates the device may be removed from the slot without prior notification.			RWO
04	PIP	Power Indicator Present: 0b = Indicates that a power indicator LED is not present for this slot.			RO
03	AIP	Attention Indicator Present: 0b = Indicates that an attention indicator LED is not present for this slot.			RO
02	MSP	MRL Sensor Present: 0b = Indicates that an MRL sensor is not present.			RO
01	PCP	Power Controller Present: 0b = Indicates that a power controller is not implemented for this slot.			RO
00	ABP	Attention Button Present: 0b = Indicates that an attention button is not implemented for this slot.			RO

12.1.1.32 Offset 58h: Slot Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-33. Offset 58h: Slot Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 58h **Offset End:** 59h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved			
12	LACE	Link Active Changed Enable — When set, this field enables generation of a hot plug interrupt when the Data Link Layer Link Active field (B0:D28:F0/F1/F2/F3:52h:bit 13) is changed.			RW
11	Reserved	Reserved			
10	PCC	Power Controller Control — This bit has no meaning for module based Hot-Plug.			RO
09 :06	Reserved	Reserved			
05	HPE	Hot Plug Interrupt Enable: 0 = Hot plug interrupts based on Hot-Plug events is disabled. 1 = Enables generation of a Hot-Plug interrupt on enabled Hot-Plug events.			RW
04	Reserved	Reserved			
03	PDE	Presence Detect Changed Enable: 0 = Hot plug interrupts based on presence detect logic changes is disabled. 1 = Enables the generation of a Hot-Plug interrupt or wake message when the presence detect logic changes state.			RW
02 :00	Reserved	Reserved			

12.1.1.33 Offset 5Ah: Slot Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-34. Offset 5Ah: Slot Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved			
08	LASC	Link Active State Changed: 1 = This bit is set when the value reported in Data Link Layer Link Active field of the Link Status register (B0:D28:F0/F1/F2/F3:52h:bit 13) is changed. In response to a Data Link Layer State Changed event, software must read Data Link Layer Link Active field of the Link Status register to determine if the link is active before initiating configuration cycles to the hot plugged device.			RWC
07	Reserved	Reserved			
06	PDS	Presence Detect State — If XCAP.SI (B0:D28:F0/F1/F2/F3:42h:bit 8) is set (indicating that this root port spawns a slot), then this bit: 0 = Indicates the slot is empty. 1 = Indicates the slot has a device connected. Otherwise, if XCAP.SI is cleared, this bit is always set (1).			RO
05	MS	MRL Sensor State — Reserved as the MRL sensor is not implemented.			
04	Reserved	Reserved			
03	PDC	Presence Detect Changed: 0 = No change in the PDS bit. 1 = The PDS bit changed states.			RWC
02	MSC	MRL Sensor Changed — Reserved as the MRL sensor is not implemented.			
01	PFD	Power Fault Detected — Reserved as a power controller is not implemented.			
00	Reserved	Reserved			

12.1.1.34 Offset 5Ch: Root Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-35. Offset 5Ch: Root Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 5Ch **Offset End:** 5Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :04	Reserved	Reserved			
03	PIE	PME Interrupt Enable: 0 = Interrupt generation disabled. 1 = Interrupt generation enabled when PCISTS.Inerrupt Status (B0:D28:F0/F1/F2/F3:60h, bit 16) is in a set state (either due to a 0 to 1 transition, or due to this bit being set with RSTS.IS already set).			RW
02	SFE	System Error on Fatal Error Enable: 0 = An SERR# will not be generated. 1 = An SERR# will be generated, assuming CMD. See (B0:D28:F0/F1/F2/F3:04, bit 8) to see if it is set and if a fatal error is reported by any of the devices in the hierarchy of this root port, including fatal errors in this root port.			RW
01	SNE	System Error on Non-Fatal Error Enable: 0 = An SERR# will not be generated. 1 = An SERR# will be generated, assuming CMD. See (B0:D28:F0/F1/F2/F3:04, bit 8) to see if it is set and if a non-fatal error is reported by any of the devices in the hierarchy of this root port, including non-fatal errors in this root port.			RW
00	SCE	System Error on Correctable Error Enable: 0 = An SERR# will not be generated. 1 = An SERR# will be generated, assuming CMD.See (B0:D28:F0/F1/F2/F3:04, bit 8) to see if a correctable error is reported by any of the devices in the hierarchy of this root port, including correctable errors in this root port.			RW

12.1.1.35 Offset 60h: Root Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-36. Offset 60h: Root Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 60h **Offset End:** 63h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved			
17	PP	PME Pending: 0 = When the original PME is cleared by software, it will be set again, the requestor ID will be updated, and this bit will be cleared. 1 = Indicates another PME is pending when the PME status bit is set.			RO
16	PS	PME Status: 0 = PME was not asserted. 1 = Indicates that PME was asserted by the requestor ID in RID. Subsequent PMEs are kept pending until this bit is cleared.			RWC
15 :00	RID	PME Requestor ID — Indicates the PCI requestor ID of the last PME requestor. Valid only when PS is set.			RO

12.1.1.36 Offset 64h: Device Capabilities 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-37. Offset 64h: Device Capabilities 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 00000016h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :05	Reserved	Reserved			
04	CTDS	Completion Timeout Disable Supported — A value of 1b indicates support for the Completion Timeout Disable mechanism.			RO
03 :00	CTRS	Completion Timeout Ranges Supported – This field indicates device support for the optional Completion Timeout programmability mechanism. This mechanism allows system software to modify the Completion Timeout value. This field is hardwired to support 10 ms to 250 ms and 250 ms to 4 s.			RO

12.1.1.37 Offset 68h: Device Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-38. Offset 68h: Device Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 68h **Offset End:** 69h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :05	Reserved	Reserved			
04	CTD	Completion Timeout Disable — When set to 1b, this bit disables the Completion Timeout mechanism. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued.			RW
03 :00	CTV	Completion Timeout Value — This field allows system software to modify the Completion Timeout value. 0000b Default range: 40-50 ms (spec range 50 us to 50 ms) 0101b: 40-50 ms (spec range is 16 ms to 55 ms) 0110b: 160-170 ms (spec range is 65 ms to 210 ms) 1001b: 400-500 ms (spec range is 260 ms to 900 ms) 1010b: 1.6-1.7 s (spec range is 1 s to 3.5 s) All other values are Reserved. Software is permitted to change the value in this field at any time. For requests already pending when the Completion Timeout Value is changed, hardware is permitted to use either the new or the old value for the outstanding requests, and is permitted to base the start time for each request either on when this value was changed or on when each request w as issued.			RW

12.1.1.38 Offset 70h: Link Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-39. Offset 70h: Link Control 2 Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 70h **Offset End:** 71h	
Size: 16 bit	**Default:** 0001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :04	Reserved	Reserved			
03 :00	TLS	Target Link Speed — This field sets an upper limit on Link operational speed by restricting the values advertised by the upstream component in its training sequences. 0001b: 2.5 GT/s Target Link Speed All other values reserved			RO

12.1.1.39 Offset 80h: Message Signaled Interrupt Identifiers Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-40. Offset 80h: Message Signaled Interrupt Identifiers Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 80h **Offset End:** 81h	
Size: 16 bit	**Default:** 9005h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Pointer — Indicates the location of the next pointer in the list.			RO
07 :00	CID	Capability ID — Capabilities ID indicates MSI.			RO

12.1.1.40 Offset 82h: Message Signaled Interrupt Message Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-41. Offset 82h: Message Signaled Interrupt Message Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	Reserved	Reserved			
07	C64	64 Bit Address Capable — Capable of generating a 32-bit message only.			RO
06 :04	MME	Multiple Message Enable — These bits are R/W for software compatibility, but only one message is ever sent by the root port.			RW
03 :01	MMC	Multiple Message Capable — Only one message is required.			RO
00	MSIE	MSI Enable: 0 = MSI is disabled. 1 = MSI is enabled and traditional interrupt pins are not used to generate interrupts. CMD.BME (B0:D28:F0/F1/F2/F3:04h:bit 2) must be set for an MSI to be generated. If CMD.BME is cleared, and this bit is set, no interrupts (not even pin based) are generated.			RW

12.1.1.41 Offset 84h: Message Signaled Interrupt Message Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-42. Offset 84h: Message Signaled Interrupt Message Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address — Lower 32 bits of the system specified message address, always DW aligned.			RW
01 :00	Reserved	Reserved			

12.1.1.42 Offset 88h: Message Signaled Interrupt Message Data Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-43. Offset 88h: Message Signaled Interrupt Message Data Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 88h **Offset End:** 89h	
Size: 16 bit	**Default:** 0000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	Data — This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD[15:0]) during the data phase of the MSI memory write transaction.			RW

12.1.1.43 Offset 90h: Subsystem Vendor Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-44. Offset 90h: Subsystem Vendor Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 90h **Offset End:** 91h	
Size: 16 bit	**Default:** A00Dh			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — Indicates the location of the next pointer in the list.			RO
07 :00	CID	Capability Identifier — Value of 0Dh indicates this is a PCI bridge subsystem vendor capability.			RO

12.1.1.44 Offset 94h: Subsystem Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-45. Offset 94h: Subsystem Vendor Identification Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	SID	Subsystem Identifier — Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).			RWO
15 :00	SVID	Subsystem Vendor Identifier — Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).			RWO

12.1.1.45 Offset A0h: Power Management Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-46. Offset A0h: Power Management Capability Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** A0h **Offset End:** A1h	
Size: 16 bit	**Default:** 0001h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — Indicates this is the last item in the list.			RO
07 :00	CID	Capability Identifier — Value of 01h indicates this is a PCI power management capability.			RO

12.1.1.46 Offset A2h: PCI Power Management Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-47. Offset A2h: PCI Power Management Capabilities Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:						
View: PCI		**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** A2h **Offset End:** A3h	
Size: 16 bit		**Default:** C802h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	:11	PMES	PME_Support — Indicates PME# is supported for states D0, $D3_{HOT}$ and $D3_{COLD}$. The root port does not generate PME#, but reporting that it does is necessary for some legacy operating systems to enable PME# in devices connected behind this root port.			RO
	10	D25	D2_Support — The D2 state is not supported.			RO
	09	D15	D1_Support — The D1 state is not supported.			RO
08	:06	AC	Aux_Current — Reports 375 mA maximum suspend well current required when in the $D3_{COLD}$ state.			RO
	05	DSI	Device Specific Initialization: 1 = Indicates that no device-specific initialization is required.			RO
	04	Reserved	Reserved			
	03	PMEC	PME Clock: 1 = Indicates that PCI clock is not required to generate PME#.			RO
02	:00	VS	Version — Indicates support for *Revision 1.1 of the PCI Power Management Specification*.			RO

12.1.1.47 Offset A4h: PCI Power Management Control and Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-48. Offset A4h: PCI Power Management Control and Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** A4h **Offset End:** A7h	
Size: 16 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23	BPCE	Bus Power / Clock Control Enable — Reserved per *PCI Express* Base Specification, Revision 1.0a*.			
22	B23S	B2/B3 Support — Reserved per *PCI Express* Base Specification, Revision 1.0a*.			
21 :16	Reserved	Reserved			
15	PMES	PME Status: 1 = Indicates a PME was received on the downstream link.			RO
14 :09	Reserved	Reserved			
08	PMEE	PME Enable: 1 = Indicates PME is enabled. The root port takes no action on this bit, but it must be R/W for some legacy operating systems to enable PME# on devices connected to this root port. This bit is sticky and resides in the resume well. The reset for this bit is RSMRST# which is not asserted during a warm reset.			RW
07 :02	Reserved	Reserved			
01 :00	PS	Power State — This field is used both to determine the current power state of the root port and to set a new power state. The values are: 00 = D0 state 11 = $D3_{HOT}$ state When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Type 1 configuration cycles are also not accepted. Interrupts are not required to be blocked as software will disable interrupts prior to placing the port into $D3_{HOT}$. If software attempts to write a '10' or '01' to these bits, the write will be ignored.			RW

12.1.1.48 Offset D4h: Miscellaneous Port Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-49. Offset D4h: Miscellaneous Port Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** D4h **Offset End:** D7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04	ASPMCOEN	ASPM Control Override Enable: 0 = Root Port will use the values in the ASPM Control Override registers 1 = Root Port will use the ASPM Registers in the Link Control register. This register allows BIOS to control the Root Port ASPM settings instead of the OS.			RW
03 :02	ASPMO	ASPM Control Override — Provides BIOS control of whether Root Port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled.			RW
01	EOIFD	EOI Forwarding Disable — When set, EOI messages are not claimed on the backbone by this port and will not be forwarded across the PCIe* link. 0 = EOI forwarding is enabled. 1 = EOI forwarding is disabled.			RW
00	LICTM	L1 Completion Timeout Mode: 0 = PCI Express* Specification Compliant. Completion timeout is disabled during software initiated L1, and enabled during ASPM initiate L1. 1 = Completion timeout is enabled during L1, regardless of how L1 entry was initiated.			RW

12.1.1.49 Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-50. Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** D8h **Offset End:** DBh	
Size: 32 bit	**Default:** 08110000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	PMCE	Power Management SCI Enable: 0 = SCI generation based on a power management event is disabled. 1 = Enables the root port to generate SCI whenever a power management event is detected.			RW
30	HPCE	Hot Plug SCI Enable: 0 = SCI generation based on a Hot-Plug event is disabled. 1 = Enables the root port to generate SCI whenever a Hot-Plug event is detected.			RW
29	LHO	Link Hold Off: 1 = Port will not take any TLP. This is used during loopback mode to fill up the downstream queue.			RW
28	ATE	Address Translator Enable — This bit is used to enable address translation via the AT bits in this register during loopback mode. 0 = Disable 1 = Enable			RW
27	LR	Lane Reversal: 0 = This register reads the setting of the PCIELR1 Soft Strap. 1 = PCI Express* Lanes 0-3 are reversed. No Lane reversal (default). - The port configuration straps must be set such that Port 1 or Port 5 is configured as a x4 port using lanes 0-3, or 4-7 when Lane Reversal is enabled. x2 lane reversal is not supported. - This register is only valid on port 1 (for ports 1-4) or port 5 (for ports 5-8).			RO
26	IRBNCE	Invalid Receive Bus Number Check Enable — When set, the receive transaction layer will signal an error if the bus number of a Memory request does not fall within the range between SCBN and SBBN. If this check is enabled and the request is a memory write, it is treated as an Unsupported Request. If this check is enabled and the request is a non-posted memory read request, the request is considered a Malformed TLP and a fatal error. Messages, I/O, Config, and Completions are never checked for valid bus number.			RW
25	IRRCE	Invalid Receive Range Check Enable — When set, the receive transaction layer will treat the TLP as an Unsupported Request error if the address range of a Memory request does not outside the range between prefetchable and non-prefetchable base and limit. Messages, I/O, Configuration, and Completions are never checked for valid address ranges.			RW

Table 12-50. Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** D8h **Offset End:** DBh	
Size: 32 bit	**Default:** 08110000h			**Power Well:**	

Bit Range	Bit Acronym	Bit Description	Sticky	Bit Reset Value	Bit Access
24	BMERCE	BME Receive Check Enable — When set, the receive transaction layer will treat the TLP as an Unsupported Request error if a memory read or write request is received and the Bus Master Enable bit is not set. Messages, IO, Config, and Completions are never checked for BME.			RW
23	Reserved	Reserved			
22	FORCEDET	Detect Override: 0 = Normal operation. Detected output from AFE is sampled for presence detection. 1 = Override mode. Ignores AFE detect output and link training proceeds as if a device were detected.			RW
21	FCDL1E	Flow Control During L1 Entr: 0 = No flow control update DLLPs sent during L1 Ack transmission. 1 = Flow control update DLLPs sent during L1 Ack transmission as required to meet the 30 μs periodic flow control update.			RW
20 :18	UCEL	Unique Clock Exit Latency — This value represents the L0s Exit Latency for unique-clock configurations (LCTL.CCC = 0) (B0:D28:F0/F1/F2/F3:Offset 50h:bit 6). It defaults to 512 ns to less than 1 μs, but may be overridden by BIOS.			RW
17 :15	CCEL	Common Clock Exit Latency — This value represents the L0s Exit Latency for common-clock configurations (LCTL.CCC = 1) (B0:D28:F0/F1/F2/F3:Offset 50h:bit 6). It defaults to 128 ns to less than 256 ns, but may be overridden by BIOS.			RW
14 :08	Reserved	Reserved			
07	PAE	Port I/OxApic Enable: 0 = Hole is disabled. 1 = A range is opened through the bridge for the following memory addresses: **Port #** / **Address** 1 / FEC1_0000h – FEC1_7FFFh 2 / FEC1_8000h – FEC1_FFFFh 3 / FEC2_0000h – FEC2_7FFFh 4 / FEC2_8000h – FEC2_FFFFh			RW
06 :03	Reserved	Reserved			

Table 12-50. Offset D8h: Miscellaneous Port Configuration Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** D8h **Offset End:** DBh	
Size: 32 bit	**Default:** 08110000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BT	Bridge Type — This register can be used to modify the Base Class and Header Type fields from the default P2P bridge to a Host Bridge. Having the root port appear as a Host Bridge is useful in some server configurations. 0 = The root port bridge type is a P2P Bridge, Header Sub-Class = 04h, and Header Type = Type 1. 1 = The root port bridge type is a P2P Bridge, Header Sub-Class = 00h, and Header Type = Type 0.			RO
01	HPME	Hot Plug SMI Enable: 0 = SMI generation based on a Hot-Plug event is disabled. 1 = Enables the root port to generate SMI whenever a Hot-Plug event is detected.			RW
00	PMME	Power Management SMI Enable: 0 = SMI generation based on a power management event is disabled. 1 = Enables the root port to generate SMI whenever a power management event is detected.			RW

12.1.1.50 Offset DCh: SMI/SCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-51. Offset DCh: SMI/SCI Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** DCh **Offset End:** DFh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	PMCS	Power Management SCI Status: 1 = PME control logic needs to generate an interrupt, and this interrupt has been routed to generate an SCI.			RWC
30	HPCS	Hot Plug SCI Status: 1 = Hot-Plug controller needs to generate an interrupt, and has this interrupt been routed to generate an SCI.			RWC
29 :05	Reserved	Reserved			
04	HPLAS	Hot Plug Link Active State Changed SMI Status: 1 = SLSTS.LASC (B0:D28:F0/F1/F2/F3:5A, bit 8) transitioned from 0-to-1, and MPC.HPME (B0:D28:F0/F1/F2/F3:D8, bit 1) is set. When this bit is set, an SMI# will be generated.			RWC
03	HPCCM	Hot Plug Command Completed SMI Status: 1 = SLSTS.CC (B0:D28:F0/F1/F2/F3:5A, bit 4) transitioned from 0-to-1, and MPC.HPME (B0:D28:F0/F1/F2/F3:D8, bit 1) is set. When this bit is set, an SMI# will be generated.			RWC
02	HPABM	Hot Plug Attention Button SMI Status: 1 = SLSTS.ABP (B0:D28:F0/F1/F2/F3:5A, bit 0) transitioned from 0-to-1, and MPC.HPME (B0:D28:F0/F1/F2/F3:D8, bit 1) is set. When this bit is set, an SMI# will be generated.			RWC
01	HPPDM	Hot Plug Presence Detect SMI Status: 1 = SLSTS.PDC (B0:D28:F0/F1/F2/F3:5A, bit 3) transitioned from 0-to-1, and MPC.HPME (B0:D28:F0/F1/F2/F3:D8, bit 1) is set. When this bit is set, an SMI# will be generated.			RWC
00	PMMS	Power Management SMI Status: 1 = RSTS.PS (B0:D28:F0/F1/F2/F3:60, bit 16) transitioned from 0-to-1, and MPC.PMME (B0:D28:F0/F1/F2/F3:D8, bit 1) is set.			RWC

12.1.1.51 Offset E1h: Root Port Dynamic Clock Gating Enable (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-52. Offset E1h: Root Port Dynamic Clock Gating Enable (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** E1h **Offset End:** E1h	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved			RO
03	SRDLCGEN	Shared Resource Dynamic Link Clock Gating Enable: 0 = Disables dynamic clock gating of the shared resource link clock domain. 1 = Enables dynamic clock gating on the root port shared resource link clock domain. Only the value from Port 1 is used for ports 1–4. Only the value from Port 5 is used for ports 5-8.			RW
02	SRDBCGEN	Shared Resource Dynamic Backbone Clock Gate: 0 = Disables dynamic clock gating of the shared resource backbone clock domain. 1 = Enables dynamic clock gating on the root port shared resource backbone clock domain. Only the value from Port 1 is used for ports 1–4. Only the value from Port 5 is used for ports 5-8.			RW
01	RPDLCGEN	Root Port Dynamic Link Clock Gate Enable: 0 = Disables dynamic clock gating of the root port link clock domain. 1 = Enables dynamic clock gating on the root port link clock domain.			RW
00	RPDBCGEN	Root Port Dynamic Backbone Clock Gate Enable: 0 = Disables dynamic clock gating of the root port backbone clock domain. 1 = Enables dynamic clock gating on the root port backbone clock domain.			RW

12.1.1.52 Offset E8h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-53. Offset E8h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** E8h **Offset End:** EBh	
Size: 32 bit	**Default:** 00000020h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved			
01	PECR1	PECR1 Field 2 — BIOS may set this bit to 1.			RW
00	Reserved	Reserved			

12.1.1.53 Offset 104h: Uncorrectable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

This register maintains its state through a platform reset. It loses its state upon suspend.

Table 12-54. Offset 104h: Uncorrectable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 00000000000x0xxx0x0x0000000x0000b			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved			
20	URE	Unsupported Request Error Status — Indicates an unsupported request was received.			RWC
19	EE	ECRC Error Status — ECRC is not supported.			RO
18	MT	Malformed TLP Status — Indicates a malformed TLP was received.			RWC
17	RO	Receiver Overflow Status — Indicates a receiver overflow occurred.			RWC
16	UC	Unexpected Completion Status — Indicates an unexpected completion was received.			RWC
15	CA	Completion Abort Status — Indicates a completer abort was received.			RWC
14	CT	Completion Timeout Status — Indicates a completion timed out. This bit is set if Completion Timeout is enabled and a completion is not returned within the time specified by the Completion TImeout Value			RWC

Table 12-54. Offset 104h: Uncorrectable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 00000000000x0xxx0x0x0000000x0000b			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
13	FCPE	Flow Control Protocol Error Status — Flow Control Protocol Errors not supported.			RO
12	PT	Poisoned TLP Status — Indicates a poisoned TLP was received.			RWC
11 :05	Reserved	Reserved			
04	DLPE	Data Link Protocol Error Status — Indicates a data link protocol error occurred.			RWC
03 :01	Reserved	Reserved			
00	TE	Training Error Status — Training Errors not supported.			RO

12.1.1.54 Offset 108h: Uncorrectable Error Mask (PCI Express*—B0:D28:F0/F1/F2/F3)

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled.

Table 12-55. Offset 108h: Uncorrectable Error Mask (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved			
20	URE	Unsupported Request Error Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
19	EE	ECRC Error Mask — ECRC is not supported.			RO
18	MT	Malformed TLP Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
17	RO	Receiver Overflow Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3/:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
16	UC	Unexpected Completion Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
15	CA	Completion Abort Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
14	CT	Completion Timeout Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
13	FCPE	Flow Control Protocol Error Mask — Flow Control Protocol Errors not supported.			RO
12	PT	Poisoned TLP Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
11 :05	Reserved	Reserved			

Table 12-55. Offset 108h: Uncorrectable Error Mask (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
04	DLPE	Data Link Protocol Error Mask: 0 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is enabled. 1 = The corresponding error in the UES register (B0:D28:F0/F1/F2/F3:144) is masked.			RWO
03 :01	Reserved	Reserved			
00	TE	Training Error Mask — Training Errors not supported			RO

12.1.1.55 Offset 10Ch: Uncorrectable Error Severity (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-56. Offset 10Ch: Uncorrectable Error Severity (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00060011h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved			
20	URE	Unsupported Request Error Severity: 0 = Error considered non-fatal. (Default) 1 = Error is fatal.			RW
19	EE	ECRC Error Severity — ECRC is not supported.			RO
18	MT	Malformed TLP Severity: 0 = Error considered non-fatal. 1 = Error is fatal. (Default)			RW
17	RO	Receiver Overflow Severity: 0 = Error considered non-fatal. 1 = Error is fatal. (Default)			RO
16	Reserved	Reserved			
15	CA	Completion Abort Severity: 0 = Error considered non-fatal. (Default) 1 = Error is fatal.			RW
14	Reserved	Reserved			
13	FCPE	Flow Control Protocol Error Severity — Flow Control Protocol Errors not supported.			RO
12	PT	Poisoned TLP Severity: 0 = Error considered non-fatal. (Default) 1 = Error is fatal.			RW
11 :05	Reserved	Reserved			
04	DLPE	Data Link Protocol Error Severity: 0 = Error considered non-fatal. 1 = Error is fatal. (Default)			RW
03 :01	Reserved	Reserved			
00	TE	Training Error Severity. TE is not supported.			RW

12.1.1.56 Offset 110h: Correctable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-57. Offset 110h: Correctable Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13	ANFES	Advisory Non-Fatal Error Status: 0 = Advisory Non-Fatal Error did not occur. 1 = Advisory Non-Fatal Error did occur.			RWC
12	RTT	Replay Timer Timeout Status — Indicates the replay timer timed out.			RWC
11 :09	Reserved	Reserved			
08	RNR	Replay Number Rollover Status — Indicates the replay number rolled over.			RWC
07	BD	Bad DLLP Status — Indicates a bad DLLP was received.			RWC
06	BT	Bad TLP Status — Indicates a bad TLP was received.			RWC
05 :01	Reserved	Reserved			
00	RE	Receiver Error Status — Indicates a receiver error occurred.			RWC

12.1.1.57 Offset 114h: Correctable Error Mask Register (PCI Express*—B0:D28:F0/F1/F2/F3)

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled.

Table 12-58. Offset 114h: Correctable Error Mask Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 00002000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13	ANFEM	Advisory Non-Fatal Error Mask: 0 = Does not mask Advisory Non-Fatal errors. 1 = Masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting. The correctable error detected bit in device status register is set whenever the Advisory Non-Fatal error is detected, independent of this mask bit.			RWC
12	RTT	Replay Timer Timeout Mask — Mask for replay timer timeout.			RWC
11 :09	Reserved	Reserved			
08	RNR	Replay Number Rollover Mask — Mask for replay number rollover.			RWC
07	BD	Bad DLLP Mask — Mask for bad DLLP reception.			RWC
06	BT	Bad TLP Mask — Mask for bad TLP reception.			RWC
05 :01	Reserved	Reserved			
00	RE	Receiver Error Mask — Mask for receiver errors.			RWC

12.1.1.58 Offset 118h: Advanced Error Capabilities and Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-59. Offset 118h: Advanced Error Capabilities and Control Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 16 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved			
08	ECE	ECRC Check Enable — ECRC is not supported.			RO
07	ECC	ECRC Check Capable — ECRC is not supported.			RO
06	EGE	ECRC Generation Enable — ECRC is not supported.			RO
05	EGC	ECRC Generation Capable — ECRC is not supported.			RO
04 :00	FEP	First Error Pointer			RO

12.1.1.59 Offset 130h: Root Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-60. Offset 130h: Root Error Status Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 130h **Offset End:** 133h	
Size: 32 bit	**Default:** 00000000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	AEMN	Advanced Error Interrupt Message Number — There is only one error interrupt allocated.			RO
26 :07	Reserved	Reserved			
06	FEMR	Fatal Error Messages Received — Set when one or more Fatal Uncorrectable Error Messages have been received.			RO
05	NFEMR	Non-Fatal Error Messages Received — Set when one or more Non-Fatal Uncorrectable error messages have been received			RO
04	FUF	First Uncorrectable Fatal — Set when the first Uncorrectable Error message received is for a fatal error.			RO
03	MENR	Multiple ERR_FATAL/NONFATAL Received — For the PCH, only one error will be captured.			RO
02	ENR	ERR_FATAL/NONFATAL Received: 0 = No error message received. 1 = Either a fatal or a non-fatal error message is received.			RWC
01	MCR	Multiple ERR_COR Received — For the PCH, only one error will be captured.			RO
00	CR	ERR_COR Received: 0 = No error message received. 1 = A correctable error message is received.			RWC

12.1.1.60 Offset 180h: Root Complex Topology Capability List Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-61. Offset 180h: Root Complex Topology Capability List Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D28:F0/F1/F2/F3		**Offset Start:** 180h **Offset End:** 183h	
Size: 32 bit	**Default:** 00010005h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	NEXT	Next Capability — Indicates the next item in the list, in this case, end of list.			RO
19 :16	CV	Capability Version — Indicates the version of the capability structure.			RO
15 :00	CID	Capability ID — Indicates this is a root complex topology capability.			RO

12.1.1.61 Offset 184h: Element Self Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-62. Offset 184h: Element Self Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)

<table>
<tr><td colspan="6">Description:</td></tr>
<tr><td>View: PCI</td><td colspan="2">BAR: Configuration</td><td>Bus:Device:Function: B0:D28:F0/F1/F2/F3</td><td colspan="2">Offset Start: 184h
Offset End: 187h</td></tr>
<tr><td>Size: 32 bit</td><td colspan="2">Default: See register description</td><td></td><td colspan="2">Power Well:</td></tr>
<tr><td>Bit Range</td><td>Bit Acronym</td><td colspan="2">Bit Description</td><td>Bit Reset Value</td><td>Bit Access</td></tr>
<tr><td>31 :24</td><td>PN</td><td colspan="2">Port Number — Indicate the ingress port number for the root port. There is a different value per port:

<table><tr><th>Port #</th><th>Value</th></tr><tr><td>1</td><td>01h</td></tr><tr><td>2</td><td>02h</td></tr><tr><td>3</td><td>03h</td></tr><tr><td>4</td><td>04h</td></tr></table></td><td></td><td>RO</td></tr>
<tr><td>23 :16</td><td>CID</td><td colspan="2">Component ID — This field returns the value of the ESD.CID field (PCH Config Space: Offset 0104h:bits 23:16) of the chip configuration section, that is programmed by platform BIOS, since the root port is in the same component as the RCRB.</td><td></td><td>RO</td></tr>
</table>

Table 12-62. Offset 184h: Element Self Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 184h **Offset End:** 187h	
Size: 32 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NLE	Number of Link Entries — The default value of 01h indicates one link entry (corresponding to the RCRB).			RO
07 :04	Reserved	Reserved			
03 :00	ET	Element Type — The default value of 0h indicates that the element type is a root port.			RO

12.1.1.62 Offset 190h: Upstream Link Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-63. Offset 190h: Upstream Link Description Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 190h **Offset End:** 193h	
Size: 16 bit	**Default:** 8086h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PN	Target Port Number — Indicates the port number of the RCRB.			RO
23 :16	TCID	Target Component ID — This field returns the value of the ESD.CID field (PCH Config Space: Offset 0104h:bits 23:16) of the chip configuration section, that is programmed by platform BIOS, since the root port is in the same component as the RCRB.			RO
15 :02	Reserved	Reserved			
01	LT	Link Type — Indicates that the link points to the PCH RCRB.			RO
00	LV	Link Valid — Indicates that this link entry is valid.			RO

12.1.1.63 Offset 198h: Upstream Link Base Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-64. Offset 198h: Upstream Link Base Address Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 198h **Offset End:** 19Fh	
Size: 64 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :32	BAU	Base Address Upper — The RCRB of the PCH lives in 32-bit space.			RO
31 :00	BAL	Base Address Lower — This field matches the RCBA register (B0:D28:F0:Offset F0h) value in the LPC bridge.			RO

12.1.1.64 Offset 300h: PCI Express* Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-65. Offset 300h: PCI Express* Configuration Register 2 (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28:F0/F1/F2/F3	**Offset Start:** 300h **Offset End:** 303h	
Size: 32 bit	**Default:** 60005007h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :22	Reserved	Reserved			
21	PCR2	PECR2 Field 1 — BIOS must set this bit to 1b.			RW
20 :00	Reserved	Reserved			

12.1.1.65 Offset 318h: PCI Express* Extended Test Mode Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-66. Offset 318h: PCI Express* Extended Test Mode Register (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 318h **Offset End:** 318h	
Size: 16 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved			
02	BAU	Scrambler Bypass Mode: 0 = Normal operation. Scrambler and descrambler are used. 1 = Bypasses the data scrambler in the transmit direction and the data de-scrambler in the receive direction. This functionality intended for debug/testing only. If bypassing scrambler with the PCH root port 1 in x4 configuration, each PCH root port must have this bit set.			RW
01 :00	Reserved	Reserved			

12.1.1.66 Offset 324h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)

Table 12-67. OFfset 324h: PCI Express* Configuration Register 1 (PCI Express*—B0:D28:F0/F1/F2/F3)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D28 :F0/F1/ F2/F3	**Offset Start:** 324h **Offset End:** 324h	
Size: 32 bit	**Default:** 14000016h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07 :00	PEC1	PEC1 Field 1 — BIOS must program this field to 40h.			RW

§ §

13.0 High Precision Event Timer Registers

The timer registers are memory-mapped in a non-indexed scheme. This allows the processor to directly access each register without having to use an index register. The timer register space is 1024 bytes. The registers are generally aligned on 64-bit boundaries to simplify implementation with IA64 processors. There are four possible memory address ranges beginning at 1) FED0_0000h, 2) FED0_1000h, 3) FED0_2000h, 4) FED0_3000h. The choice of address range will be selected by configuration bits in the High Precision Timer Configuration Register (Chipset Config Registers = RCBA; Offset 3404h).

13.1 High Precision Behavioral Rules

Behavioral Rules:

- Software must not attempt to read or write across register boundaries. For example, a 32-bit access should be to offset x0h, x4h, x8h, or xCh. 32-bit accesses should not be to 01h, 02h, 03h, 05h, 06h, 07h, 09h, 0Ah, 0Bh, 0Dh, 0Eh, or 0Fh. Any accesses to these offsets will result in an unexpected behavior, and may result in a master abort. However, these accesses should not result in system hangs. 64-bit accesses can only be to x0h and must not cross 64-bit boundaries.
- Software should not write to read-only registers.
- Software should not expect any particular or consistent value when reading reserved registers or bits.

13.1.1 Memory Mapped Registers

Table 13-1. Memory Mapped Registers

Offset Start	Offset End	Register ID - Description	Default Value
00h	07h	"Offset 00h: General Capabilities and Identification Register" on page 682	0429B17F8086A201h
010h	01Fh	"Offset 010h: General Configuration Register" on page 683	00000000 00000000h
020h	027h	"Offset 020h: General Interrupt Status Register" on page 684	00000000 00000000h
0F0h	0F7h	"Offset 0F0h: Main Counter Value Register" on page 685	N/A
100h	107h	"Offset 100h: Timer n Configuration and Capabilities Register" on page 687	N/A
108h	10Fh	"Offset 108h: Timer n Comparator Value Register" on page 691	N/A

13.1.1.1 Offset 00h: General Capabilities and Identification Register

Table 13-2. Offset 00h: General Capabilities and Identification Register

Description:					
View: IA F	**BAR:** HPTC		**Bus:Device:Function :**	**Offset Start:** 00h **Offset End:** 07h	
Size: 64 bit	**Default:** 0429B17F8086A201h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63:32	COUNTER_CLK_PER_CAP	Main Counter Tick Period: • This field indicates the period at which the counter increments in femptoseconds (10^-15 seconds). • This will return 0429B17F when read. • This indicates a period of 69841279 fs (69.841279 ns).			RO
31:16	VENDOR_ID_CAP	Vendor ID Capability: This is a 16-bit value assigned to Intel (8086h)			RO
15	LEG_RT_CAP	Legacy Replacement Rout Capable: Hardwired to 1. Legacy Replacement Interrupt Rout option is supported.			RO
14	Reserved	Reserved. This bit returns 0 when read.			
13	COUNT_SIZE_CAP	Counter Size Capability: Hardwired to 1. Counter is 64-bit wide.			RO
12:08	NUM_TIM_CAP	Number of Timer Capability: This field indicates the number of timers in this block. 07h = Eight timers.			RO
07:00	REV_ID	Revision Identification: This indicates which revision of the function is implemented. Default value will be 01h.			RO

13.1.1.2 Offset 010h: General Configuration Register

Table 13-3. Offset 010h: General Configuration Register

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 010h **Offset End:** 017h	
Size: 64 bit	**Default:** 00000000 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
63:02	Reserved	Reserved. These bits return 0 when read.				
01	LEG_RT_CNF	Legacy Replacement Rout: If the ENABLE_CNF bit and the LEG_RT_CNF bit are both set, then the interrupts will be routed as follows: • Timer 0 is routed to IRQ0 in 8259 or IRQ2 in the I/O APIC • Timer 1 is routed to IRQ8 in 8259 or IRQ8 in the I/O APIC • Timer 2-n is routed as per the routing in the timer n config registers. • If the Legacy Replacement Rout bit is set, the individual routing bits for Timers 0 and 1 (APIC) will have no impact. • If the Legacy Replacement Rout bit is not set, the individual routing bits for each of the timers are used. This bit will default to 0. BIOS can set it to 1 to enable the legacy replacement routing, or 0 to disable the legacy replacement routing.				RW
00	ENABLE_CNF	Overall Enable: This bit must be set to enable any of the timers to generate interrupts. If this bit is 0, then the main counter will halt (will not increment) and no interrupts will be caused by any of these timers. For level-triggered interrupts, if an interrupt is pending when the ENABLE_CNF bit is changed from 1 to 0, the interrupt status indications (in the various Txx_INT_STS bits) will not be cleared. Software must write to the Txx_INT_STS bits to clear the interrupts. ***Note:*** This bit will default to 0. BIOS can set it to 1 or 0				RW

13.1.1.3 Offset 020h: General Interrupt Status Register

Table 13-4. Offset 020h: General Interrupt Status Register

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 020h **Offset End:** 027h	
Size: 64 bit	**Default:** 00000000 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
63:08	Reserved	Reserved. These bits return 0 when read.				
07	T07_INT_STS	Timer 7 Interrupt Active — Same functionality as Timer 0.				RW
06	T06_INT_STS	Timer 6 Interrupt Active — Same functionality as Timer 0.				RW
05	T05_INT_STS	Timer 5 Interrupt Active — Same functionality as Timer 0.				RW
04	T04_INT_STS	Timer 4 Interrupt Active — Same functionality as Timer 0.				RW
03	T03_INT_STS	Timer 3 Interrupt Active — Same functionality as Timer 0.				RW
02	T02_INT_STS	Timer 2 Interrupt Active — Same functionality as Timer 0.				RW
01	T01_INT_STS	Timer 1 Interrupt Active — Same functionality as Timer 0.				RW
00	T00_INT_STS	Timer 0 Interrupt Active — • The functionality of this bit depends on whether the edge or level-triggered mode is used for this timer. If set to level-triggered mode: • This bit defaults to 0. • This bit will be set by hardware if the corresponding timer interrupt is active. • Once the bit is set, it can be cleared by software by writing a 1 to the same bit position. • Writes of 0 to this bit will have no effect. If set to edge-triggered mode: • This bit should be ignored by software. • Software should always write 0 to this bit. ***Note:*** Defaults to 0. In edge triggered mode, this bit will always read as 0 and writes will have no effect.				RWC

13.1.1.4 Offset 0F0h: Main Counter Value Register

Table 13-5. Offset 0F0h: Main Counter Value Register

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 0F0h **Offset End:** 0F7h	
Size: 64 bit	**Default:** N/A				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
63:00	COUNTER_VAL[63:0]	Counter Value: Reads return the current value of the counter. Writes load the new value to the counter. ***Notes:*** • Writes to this register should only be done while the counter is halted. • Reads to this register return the current value of the main counter. • 32-bit counters will always return 0 for the upper 32-bits of this register. • If 32-bit software attempts to read a 64-bit counter, it should first halt the counter. Since this delays the interrupts for all of the timers, this should be done only if the consequences are understood. It is strongly recommended that 32-bit software only operate the timer in 32-bit mode. • Reads to this register are monotonic. No two consecutive reads return the same value. The second of two reads always returns a larger value (unless the timer has rolled over to 0).			RW

13.1.1.5 Offset 100h: Timer n Configuration and Capabilities Register

Address Offset: Timer 0: 100–107h,
Timer 1: 120–127h,
Timer 2: 140–147h,
Timer 3: 160–167h,
Timer 4: 180–187h,
Timer 5: 1A0–1A7h,
Timer 6: 1C0–1C7h,
Timer 7: 1E0–1E7h,

The letter n can be 0, 1, 2, 3, 4, 5, 6, or 7 referring to Timer 0, 1, 2, 3, 4, 5, 6, or 7.

Reads or writes to unimplemented timers should not be attempted. Read from any unimplemented registers will return an undetermined value.

Table 13-6. Offset 100h: Timer n Configuration and Capabilities Register (Sheet 1 of 4)

Description:						
View: IA F		**BAR:** HPTC	**Bus:Device:Function:**		**Offset Start:** 100h **Offset End:** 107h	
Size: 64 bit		**Default:** N/A			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63	:56	Reserved	Reserved. These bits will return 0 when read.			
55	:52,44,43	TIMERn_INT_ROUT_CAP	Timer Interrupt Rout Capability: • In general, a 32-bit field of this register, Bits[63:32] is allocated to indicate which interrupts (IRQ[31:0]) in the 8259 or I/O(x) APIC this timer's interrupt can be routed to. • The Bits are mapped as follows: — Bit[32] = IRQ[0] — .. — Bit[63] = IRQ[31] These bits [55-52, 44, 43) are indicated here because they are the interrupts supported by the Timers. Timer 0, 1 (if LEG_RT_CNF = 0): • Bits 52, 53, 54, and 55 in this field (corresponding to IRQ 20, 21, 22, and 23) have a value of 1 to indicate the supported interrupts by this Timer. Writes will have no effect. Timer 2: • Bits 43, 52, 53, 54, and 55 in this field (corresponding to IRQ 11, 20, 21, 22, and 23) have a value of 1 to indicate the supported interrupts by this Timer. Writes will have no effect. Timer 3: • Bits 44, 52, 53, 54, and 55 in this field (corresponding to IRQ 12, 20, 21, 22, and 23) have a value of to indicate the supported interrupts by this Timer1. Writes will have no effect. Timer 4 - 7: • This field is always 0 as interrupts from these timers can only be delivered via direct interrupt messages. ***Note:*** If IRQ 11 is used for HPET #2, software should ensure IRQ 11 is not shared with any other devices to ensure the proper operation of HPET #2. • If IRQ 12 is used for HPET #3, software should ensure IRQ 12 is not shared with any other devices to ensure the proper operation of HPET #3.			RO
51	:45	Reserved	Reserved.			
42	:14	Reserved	Reserved.			

Table 13-6. Offset 100h: Timer n Configuration and Capabilities Register (Sheet 2 of 4)

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 100h **Offset End:** 107h	
Size: 64 bit	**Default:** N/A				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
13:09	TIMERn_INT_ROUT_CNF	Timer Interrupt Rout Configuration: This 5-bit binary field, decodes into a decimal number [0 - 31], to indicate the routing for the interrupt to the 8259 or I/O (x) APIC. Software writes to this field to select which interrupt in the 8259 or I/O (x) will be used for this timer's interrupt. If the value is not supported by this particular timer, then the value read back will not match what is written. The software must only write valid values. ***Note:*** For example, (10100b = 20 decimal) to indicate the routing of the interrupt to IRQ[20] Timer 4, 5, 6, 7: This field is Read-only and reads will return 0. ***Notes:*** • If the interrupt is handled using the 8259, only interrupts 0-15 are applicable and valid. Software must not program any value other than 0-15 in this field. • If the Legacy Replacement Rout bit is set, then Timers 0 and 1 will have a different routing, and this bit field has no effect for those two timers. Timer 0,1: • Software is responsible to make sure it programs a valid value (**decimal** 20, 21, 22, or 23) for this field. The PCH logic does not check the validity of the value written. Timer 2: • Software is responsible to make sure it programs a valid value (**decimal** 11, 20, 21, 22, or 23) for this field. The PCH logic does not check the validity of the value written. Timer 3: • Software is responsible to make sure it programs a valid value (**decimal** 12, 20, 21, 22, or 23) for this field. The PCH logic does not check the validity of the value written.			RW

Table 13-6. Offset 100h: Timer n Configuration and Capabilities Register (Sheet 3 of 4)

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 100h **Offset End:** 107h	
Size: 64 bit	**Default:** N/A				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
08	TIMERn_32Mode_CNF	**Timer n 32-bit Mode Configuration:** Software can set this bit to force a 64-bit timer to behave as a 32-bit timer. Timer 0: • Bit is read/write (default to 0) • 0 = 64 bit; 1 = 32 bit Timers 1, 2, 3, 4, 5, 6, 7: • Hardwired to 0. Writes have no effect (since these two timers are 32-bits). ***Note:*** When this bit is set to 1, the hardware counter will do a 32-bit operation on comparator match and rollovers; thus, the upper 32-bit of the Timer 0 Comparator Value register is ignored. The upper 32-bit of the main counter is not involved in any rollover from lower 32-bit of the main counter and becomes all zeros				RW/RO
07	Reserved	Reserved. This bit returns 0 when read.				
06	TIMERn_VAL_SET_CNF	Timer n Value Set: Software uses this bit only for Timer 0 if it has been set to periodic mode. By writing this bit to a 1, the software is then allowed to directly set the timer's accumulator. Software does **not** have to write this bit back to 1 (it automatically clears). Software should not write a 1 to this bit position if the timer is set to non-periodic mode. ***Note:*** This bit will return 0 when read. Writes will only have an effect for Timer 0 if it is set to periodic mode. Writes will have no effect for Timers 1, 2, 3, 4, 5, 6, 7.				RW
05	TIMERn_SIZE_CAP	Timer n Size: This read only field indicates the size of the timer. Timer 0: • Value is 1 (64-bits). Timers 1, 2, 3, 4, 5, 6, 7.: • Value is 0 (32-bits).				RO
04	TIMERn_PER_INT_CAP	Periodic Interrupt Capable: If this bit is 1, the hardware supports a periodic mode for this timer's interrupt. Timer 0: • Hardwired to 1 (supports the periodic interrupt). Timers 1, 2, 3, 4, 5, 6, 7.: • Hardwired to 0 (does not support periodic interrupt).				RO

Table 13-6. Offset 100h: Timer n Configuration and Capabilities Register (Sheet 4 of 4)

Description:						
View: IA F		**BAR:** HPTC	**Bus:Device:Function:**		**Offset Start:** 100h **Offset End:** 107h	
Size: 64 bit		**Default:** N/A			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	03	TIMERn_TYPE_CNF	Timer n Type: Timer 0: • Bit is read/write. • 0 = Disable timer to generate periodic interrupt; • 1 = Enable timer to generate a periodic interrupt. Timers 1, 2, 3, 4, 5, 6, 7.: • Hardwired to 0. Writes have no affect.			RW/RO
	02	TIMERn_INT_ENB_CNF	Timer n Interrupt Enable: This bit must be set to enable timer n to cause an interrupt when it times out. • 0 = Disable (Default). The timer can still count and generate appropriate status bits, but will not cause an interrupt. • 1 = Enable.			RW
	01	TIMERn_INT_TYPE_CNF	Timer Interrupt Type: Timers 1, 2, 3 • 0 = The timer interrupt is edge triggered. This means that an edge-type interrupt is generated. If another interrupt occurs, another edge will be generated. • 1 = The timer interrupt is level triggered. This means that a level-triggered interrupt is generated. The interrupt will be held active until it is cleared by writing to the bit in the General Interrupt Status Register. If another interrupt occurs before the interrupt is cleared, the interrupt will remain active. Timers 4, 5, 6, 7: • This bit is Read-Only, and will return 0 when read			RW
	00	Reserved	Reserved. These bits will return 0 when read.			
Note: Reads and writes to unimplemented timers should not be attempted. Read from any unimplemented registers will return an undetermined value						

13.1.1.6 Offset 108h: Timer n Comparator Value Register

Address Offset: Timer 0: 108h – 10Fh,
Timer 1: 128h – 12Fh,
Timer 2: 148h – 14Fh,
Timer 3: 168h – 16Fh,
Timer 4: 188h – 18Fh,
Timer 5: 1A8h – 1AFh,
Timer 6: 1C8h – 1CFh,
Timer 7: 1E8h – 1EFh

Table 13-7. Offset 108h: Timer n Comparator Value Register

Description:						
View: IA F	**BAR:** HPTC		**Bus:Device:Function:**		**Offset Start:** 108h **Offset End:** 10Fh	
Size: 64 bit	**Default:** N/A				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
63:00	TIMERn_COMP_VAL	Timer Compare Value: Reads to this register return the current value of the comparator. Timers 1, 2, 3 4, 5, 6, 7 (4, 5, 6, 7) are configured to non-periodic mode: • Writes to this register load the value against which the main counter should be compared for this timer. • When the main counter equals the value last written to this register, the corresponding interrupt can be generated (if so enabled). • The value in this register does not change based on the interrupt being generated. Timer 0 is configured to periodic mode: • When the main counter equals the value last written to this register, the corresponding interrupt can be generated (if so enabled). • After the main counter equals the value in this register, the value in this register is increased by the value last written to the register. For example, if the value written to the register is 00000123h, then: 1. An interrupt will be generated when the main counter reaches 00000123h 2. The value in this register will then be adjusted by the hardware to 00000246h. 3. Another interrupt will be generated when the main counter reaches 00000246h. 4. The value in this register will then be adjusted by the hardware to 00000369h • As each periodic interrupt occurs, the value in this register will increment. When the incremented value is greater than the maximum value possible for this register (FFFFFFFFh for a 32-bit timer or FFFFFFFFFFFFFFFFh for a 64-bit timer), the value will wrap around through 0. For example, if the current value in a 32-bit timer is FFFF0000h and the last value written to this register is 20000, then after the next interrupt the value will change to 00010000h Default value for each timer is all 1s for the bits that are implemented. For example, a 32-bit timer has a default value of 00000000FFFFFFFFh. A 64-bit timer has a default value of FFFFFFFFFFFFFFFFh.			RW

§ §

14.0 Serial Peripheral Interface (SPI)

The Serial Peripheral Interface resides in memory mapped space. This function contains registers that allow for the setup and programming of devices that reside on the SPI interface.

Note: All registers in this function (including memory-mapped registers) must be addressable in byte, word, and dword quantities. The software must always make register accesses on natural boundaries (for example, DWord accesses must be on dword boundaries; word accesses on word boundaries, etc.) In addition, the memory-mapped register space must not be accessed with the LOCK semantic exclusive-access mechanism. If software attempts exclusive-access mechanisms to the SPI memory-mapped space, the results are undefined.

14.1 Serial Peripheral Interface Memory Mapped Configuration Registers

The SPI Host Interface registers are memory-mapped in the RCRB (Root Complex Register Block) Chipset Register Space with a base address (SPIBAR) of 3800h and are located within the range of 3800h to 39FFh. The address for RCRB can be found in RCBA Register. The individual registers are then accessible at SPIBAR + Offset as indicated in the following table.

These memory mapped registers must be accessed in byte, word, or dword quantities.

14.1.1 Serial Peripheral Interface (SPI) Register Address Map

Table 14-1. Serial Peripheral Interface (SPI) Register Address Map (Sheet 1 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: BIOS Flash Primary Region Register (SPI Memory Mapped Configuration Registers)" on page 694	00000000h
04h	05h	"Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers)" on page 695	0000h
06h	07h	"Offset 06h: Hardware Sequencing Flash Control Register SPI Memory Mapped Configuration Registers)" on page 697	0000h
08h	0Bh	"Offset 08h: Flash Address Register (SPI Memory Mapped Configuration Registers)" on page 698	00000000h
14h + (4*N)	17h + (4*N)	"Offset 14h: Flash Data [N] Register (SPI Memory Mapped Configuration Registers)" on page 699	00000000h
50h	53h	"Offset 50h: Flash Regions Access Permissions Register (SPI Memory Mapped Configuration Registers)" on page 700	00000202h
54h	57h	"Offset 54h: Flash Region 0 (Flash Descriptor) Register (SPI Memory Mapped Configuration Registers)" on page 701	00000000h

Table 14-1. Serial Peripheral Interface (SPI) Register Address Map (Sheet 2 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
58h	5Bh	"Offset 58h: Offset 00h: Flash Region 1 (BIOS Descriptor) Register (SPI Memory Mapped Configuration Registers)" on page 701	00000000h
5Ch	5Fh	"Offset 5Ch: Flash Region 2 (Intel ME) Register (SPI Memory Mapped Configuration Registers)" on page 702	00000000h
60h	63h	"Offset 60h: Flash Region 3 Register (SPI Memory Mapped Configuration Registers)" on page 702	00000000h
64h	67h	"Offset 64h: Flash Region 4 (Platform Data) Register (SPI Memory Mapped Configuration Registers)" on page 703	00000000h
74h	77h	"Offset 74h: Protected Range 0 Register (SPI Memory Mapped Configuration Registers)" on page 703	00000000h
78h	7Bh	"Offset 78h: Protected Range 1 Register (SPI Memory Mapped Configuration Registers)" on page 704	00000000h
7Ch	7Fh	"Offset 7Ch: Protected Range 2 Register (SPI Memory Mapped Configuration Registers)" on page 705	00000000h
80h	83h	"Offset 80h: Protected Range 3 Register (SPI Memory Mapped Configuration Registers)" on page 706	00000000h
84h	87h	"Offset 84h: Protected Range 4 Register (SPI Memory Mapped Configuration Registers)" on page 707	00000000h
90h	90h	"Offset 90h: Software Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers)" on page 708	00h
91h	93h	"Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers)" on page 709	F80000h
94h	95h	"Offset 94h: Prefix Opcode Configuration Register (SPI Memory Mapped Configuration Registers)" on page 711	0000h
96h	97h	"Offset 96h: Opcode Type Configuration Register (SPI Memory Mapped Configuration Registers)" on page 712	0000h
98h	9Fh	"Offset 98h: Opcode Menu Configuration Register (SPI Memory Mapped Configuration Registers)" on page 713	0000000000000000h
A0h	A3h	"Offset A0h: BIOS Base Address Configuration Register (SPI Memory Mapped Configuration Registers)" on page 714	00000000h
B0h	B3h	"Offset B0h: Flash Descriptor Observability Control Register (SPI Memory Mapped Configuration Registers)" on page 715	00000000h
B4h	B7h	"Offset B4h: Flash Descriptor Observability Data Register SPI Memory Mapped Configuration Registers)" on page 715	00000000h
C0h	C3h	"Offset C0h: Additional Flash Control Register (SPI Memory Mapped Configuration Registers)" on page 716	00000000h
C4h	C7h	"Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers)" on page 717	00000000h
C0h	C3h	"Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers)" on page 719	00000000h
D0h	D3h	"Offset D0h: Flash Partition Boundary (SPI Memory Mapped Configuration Registers)" on page 721	00000000h

14.1.1.1 Offset 00h: BIOS Flash Primary Region Register (SPI Memory Mapped Configuration Registers)

Note: This register is only applicable when SPI device is in descriptor mode.

Table 14-2. Offset 00h: BIOS Flash Primary Region Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:29	Reserved	Reserved				O
28:16	PRL	BIOS Flash Primary Region Limit — This specifies address bits 24:12 for the Primary Region Limit. The value in this register loaded from the contents in the Flash Descriptor.FLREG1.Region Limit				RO
15:13	Reserved	Reserved				O
12:00	PRB	BIOS Flash Primary Region Base — This specifies address bits 24:12 for the Primary Region Base The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Base				RO

14.1.1.2 Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers)

Table 14-3. Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
15	FLOCKDN	Flash Configuration Lock-Down — When set to 1, those Flash Program Registers that are locked down by this FLOCKDN bit cannot be written. Once set to 1, this bit can only be cleared by a hardware reset due to a global reset or host partition reset in an Intel® ME enabled system.				RWL
14	FDV	Flash Descriptor Valid — This bit is set to a 1 if the Flash Controller read the correct Flash Descriptor Signature. If the Flash Descriptor Valid bit is not 1, software cannot use the Hardware Sequencing registers, but must use the software sequencing registers. Any attempt to use the Hardware Sequencing registers will result in the FCERR bit being set.				RO
13	FDOPSS	Flash Descriptor Override Pin-Strap Status — This register reflects the value the Flash Descriptor Override Pin-Strap. 0 = The Flash Descriptor Override strap is set 1 = No override				RO
12:06	Reserved	Reserved				O
05	SCIP	SPI Cycle In Progress — Hardware sets this bit when software sets the Flash Cycle Go (FGO) bit in the Hardware Sequencing Flash Control register. This bit remains set until the cycle completes on the SPI interface. Hardware automatically sets and clears this bit so that software can determine when read data is valid and/or when it is safe to begin programming the next command. Software must only program the next command when this bit is 0. This field is only applicable when in Descriptor mode and Hardware sequencing is being used.				RO
04:03	BERASE	Block/Sector Erase Size — This field identifies the erasable sector size for all Flash components. Valid Bit Settings: 00 = 256 Byte 01 = 4 K Byte 10 = 8 K Byte 11 = 64 K Byte If the FLA is less than FPBA then this field reflects the value in the LVSCC.LBES register. If the FLA is greater or equal to FPBA then this field reflects the value in the UVSCC.UBES register. This field is only applicable when in Descriptor mode and Hardware sequencing is being used.				RO

Table 14-3. Offset 04h: Hardware Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers) (Sheet 2 of 2)

Description:						
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit		**Default:** 0000h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	02	AEL	Access Error Log — Hardware sets this bit to a 1 when an attempt was made to access the BIOS region using the direct access method or an access to the BIOS Program Registers that violated the security restrictions. This bit is simply a log of an access security violation. This bit is cleared by software writing a 1. This field is only applicable when in Descriptor mode and Hardware sequencing is being used.			RWC
	01	FCERR	Flash Cycle Error — Hardware sets this bit to 1 when an program register access is blocked to the FLASH due to one of the protection policies or when any of the programmed cycle registers is written while a programmed access is already in progress. This bit remains asserted until cleared by software writing a 1 or until hardware reset occurs due to a global reset or host partition reset in an Intel® ME enabled system. Software must clear this bit before setting the FLASH Cycle GO bit in this register. This field is only applicable when in Descriptor mode and Hardware sequencing is being used.			RWC
	00	FDONE	Flash Cycle Done — The PCH sets this bit to 1 when the SPI Cycle completes after software previously set the FGO bit. This bit remains asserted until cleared by software writing a 1 or hardware reset due to a global reset or host partition reset in an Intel® ME enabled system. When this bit is set and the SPI SMI# Enable bit is set, an internal signal is asserted to the SMI# generation block. Software must make sure this bit is cleared prior to enabling the SPI SMI# assertion for a new programmed access. This field is only applicable when in Descriptor mode and Hardware sequencing is being used.			RWC

14.1.1.3 Offset 06h: Hardware Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers)

Table 14-4. Offset 06h: Hardware Sequencing Flash Control Register SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
15	FSMIE	Flash SPI SMI# Enable — When set to 1, the SPI asserts an SMI# request whenever the Flash Cycle Done bit is 1.				RW
14	Reserved	Reserved				O
13:08	FDBC	Flash Data Byte Count — This field specifies the number of bytes to shift in or out during the data portion of the SPI cycle. The contents of this register are 0s based with 0b representing 1 byte and 111111b representing 64 bytes. The number of bytes transferred is the value of this field plus 1. This field is ignored for the Block Erase command.				RW
07:03	Reserved	Reserved				RO
02:01	FCYCLE	Flash Cycle — This field defines the Flash SPI cycle type generated to the FLASH when the FGO bit is set as defined below: 00 = Read (1 up to 64 bytes by setting FDBC) 01 = Reserved 10 = Write (1 up to 64 bytes by setting FDBC) 11 = Block Erase				RW
00	FGO	Flash Cycle Go — A write to this register with a 1 in this bit initiates a request to the Flash SPI Arbiter to start a cycle. This register is cleared by hardware when the cycle is granted by the SPI arbiter to run the cycle on the SPI bus. When the cycle is complete, the FDONE bit is set. Software is forbidden to write to any register in the HSFLCTL register between the FGO bit getting set and the FDONE bit being cleared. Any attempt to violate this rule will be ignored by hardware. Hardware allows other bits in this register to be programmed for the same transaction when writing this bit to 1. This saves an additional memory write. This bit always returns 0 on reads.				RWS

14.1.1.4 Offset 08h: Flash Address Register (SPI Memory Mapped Configuration Registers)

Table 14-5. Offset 08h: Flash Address Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 08h **Offset End:** 0Bh	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
32	:25	Reserved	Reserved				O
24	:00	FLA	Flash Linear Address — The FLA is the starting byte linear address of a SPI Read or Write cycle or an address within a Block for the Block Erase command. The Flash Linear Address must fall within a region for which BIOS has access permissions. Hardware must convert the FLA into a Flash Physical Address (FPA) before running this cycle on the SPI bus.				RW

14.1.1.5 Offset 10h: Flash Data 0 Register (SPI Memory Mapped Configuration Registers)

14.1.1.6 Offset 14h: Flash Data [N] Register (SPI Memory Mapped Configuration Registers)

SPIBAR + 14h
SPIBAR + 18h
SPIBAR + 1Ch
SPIBAR + 20h
SPIBAR + 24h
SPIBAR + 28h
SPIBAR + 2Ch
SPIBAR + 30h
SPIBAR + 34h
SPIBAR + 38h
SPIBAR + 3Ch
SPIBAR + 40h
SPIBAR + 44h
SPIBAR + 48h
SPIBAR + 4Ch

Table 14-6. Offset 14h: Flash Data [N] Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 14h + (4*N) **Offset End:** 17h + (4*N)	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:00	FD[N]	Flash Data [N] — Similar definition as Flash Data 0. However, this register does not begin shifting until FD[N-1] has completely shifted in/out.				RW

14.1.1.7 Offset 50h: Flash Regions Access Permissions Register (SPI Memory Mapped Configuration Registers)

Table 14-7. Offset 50h: Flash Regions Access Permissions Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** 50h **Offset End:** 53h	
Size: 32 bit		**Default:** 00000202h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:24	BMWAG	BIOS Master Write Access Grant — Each bit [31:29] corresponds to Master[7:0]. BIOS can grant one or more masters write access to the BIOS region 1 overriding the permissions in the Flash Descriptor. Master[1] is Host processor/BIOS, Master[2] is Intel® Management Engine, Master[3] is Host processor/GbE. Master[0] and Master[7:4] are reserved. The contents of this register are locked by the FLOCKDN bit.				RW
23	:16	BMRAG	BIOS Master Read Access Grant — Each bit [28:16] corresponds to Master[7:0]. BIOS can grant one or more masters read access to the BIOS region 1 overriding the read permissions in the Flash Descriptor. Master[1] is Host processor/BIOS, Master[2] is Intel® Management Engine, Master[3] is Host processor/GbE. Master[0] and Master[7:4] are reserved. The contents of this register are locked by the FLOCKDN bit				RW
15	:08	BRWA	BIOS Region Write Access — Each bit [15:8] corresponds to Regions [7:0]. If the bit is set, this master can erase and write that particular region through register accesses. The contents of this register are that of the Flash Descriptor. Flash Master 1 Master Region Write Access OR a particular master has granted BIOS write permissions in their Master Write Access Grant register or the Flash Descriptor Security Override strap is set.				RO
07	:00	BRRA	BIOS Region Read Access — Each bit [7:0] corresponds to Regions [7:0]. If the bit is set, this master can read that particular region through register accesses. The contents of this register are that of the Flash Descriptor.Flash Master 1.Master Region Write Access OR a particular master has granted BIOS read permissions in their Master Read Access Grant register or the Flash Descriptor Security Override strap is set.				RO

14.1.1.8 Offset 54h: Flash Region 0 (Flash Descriptor) Register (SPI Memory Mapped Configuration Registers)

Table 14-8. Offset 54h: Flash Region 0 (Flash Descriptor) Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:29	Reserved	Reserved				O
28	:16	RL	Region Limit — This specifies address bits 24:12 for the Region 0 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG0.Region Limit				RO
15	:13	Reserved	Reserved				O
12	:00	RB/FDBAR	Region Base/Flash Descriptor Base Address Region — This specifies address bits 24:12 for the Region 0 Base The value in this register is loaded from the contents in the Flash Descriptor.FLREG0.Region Base				RO

14.1.1.9 Offset 58h: Offset 00h: Flash Region 1 (BIOS Descriptor) Register (SPI Memory Mapped Configuration Registers)

Table 14-9. Offset 58h: Offset 00h: Flash Region 1 (BIOS Descriptor) Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 58h **Offset End:** 5Bh	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:29	Reserved	Reserved				O
28	:16	RL	Region Limit — This specifies address bits 24:12 for the Region 1 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Limit				RO
15	:13	Reserved	Reserved				O
12	:00	RB	Region Base — This specifies address bits 24:12 for the Region 1 Base The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Base				RO

14.1.1.10 Offset 5Ch: Flash Region 2 (Intel ME) Register (SPI Memory Mapped Configuration Registers)

Table 14-10. Offset 5Ch: Flash Region 2 (Intel ME) Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 5Ch **Offset End:** 5Fh	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:29	Reserved	Reserved				O
28	:16	RL	Region Limit — This specifies address bits 24:12 for the Region 2 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG2.Region Limit				RO
15	:13	Reserved	Reserved				O
12	:00	RB	Region Base — This specifies address bits 24:12 for the Region 2 Base The value in this register is loaded from the contents in the Flash Descriptor.FLREG2.Region Base				RO

14.1.1.11 Offset 60h: Flash Region 3 Register (SPI Memory Mapped Configuration Registers)

Table 14-11. Offset 60h: Flash Region 3 Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 60h **Offset End:** 63h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:00	Reserved	Reserved				O

14.1.1.12 Offset 64h: Flash Region 4 (Platform Data) Register (SPI Memory Mapped Configuration Registers)

Table 14-12. Offset 64h: Flash Region 4 (Platform Data) Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:00	Reserved	Reserved				O

14.1.1.13 Offset 74h: Protected Range 0 Register (SPI Memory Mapped Configuration Registers)

Table 14-13. Offset 74h: Protected Range 0 Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 74h **Offset End:** 77h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
	31	WPE	Write Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
30	:29	Reserved	Reserved				O
28	:16	PRL	Protected Range Limit — This field corresponds to FLA address bits 24:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.				RW
	15	RPE	Read Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
14	:13	Reserved	Reserved				O
12	:00	PRB	Protected Range Base — This field corresponds to FLA address bits 24:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.				RW

14.1.1.14 Offset 78h: Protected Range 1 Register (SPI Memory Mapped Configuration Registers)

Table 14-14. Offset 78h: Protected Range 1 Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	WPE	Write Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
30:29	Reserved	Reserved				O
28:16	PRL	Protected Range Limit — This field corresponds to FLA address bits 24:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.				RW
15	RPE	Read Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
14:13	Reserved	Reserved				O
12:00	PRB	Protected Range Base — This field corresponds to FLA address bits 24:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.				RW

14.1.1.15 Offset 7Ch: Protected Range 2 Register (SPI Memory Mapped Configuration Registers)

Table 14-15. Offset 7Ch: Protected Range 2 Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 7Ch **Offset End:** 7Fh	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	WPE	Write Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
30:29	Reserved	Reserved				O
28:16	PRL	Protected Range Limit — This field corresponds to FLA address bits 24:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.				RW
15	RPE	Read Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
14:13	Reserved	Reserved				O
12:00	PRB	Protected Range Base — This field corresponds to FLA address bits 24:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.				RW

14.1.1.16 Offset 80h: Protected Range 3 Register (SPI Memory Mapped Configuration Registers)

Table 14-16. Offset 80h: Protected Range 3 Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 80h **Offset End:** 83h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	WPE	Write Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
30:29	Reserved	Reserved				O
28:16	PRL	Protected Range Limit — This field corresponds to FLA address bits 24:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.				RW
15	RPE	Read Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
14:13	Reserved	Reserved				O
12:00	PRB	Protected Range Base — This field corresponds to FLA address bits 24:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.				RW

14.1.1.17 Offset 84h: Protected Range 4 Register (SPI Memory Mapped Configuration Registers)

Table 14-17. Offset 84h: Protected Range 4 Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	WPE	Write Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
30:29	Reserved	Reserved				O
28:16	PRL	Protected Range Limit — This field corresponds to FLA address bits 24:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.				RW
15	RPE	Read Protection Enable — When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.				RW
14:13	Reserved	Reserved				O
12:00	PRB	Protected Range Base — This field corresponds to FLA address bits 24:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.				RW

14.1.1.18 Offset 90h: Software Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers)

Table 14-18. Offset 90h: Software Sequencing Flash Status Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 90h **Offset End:** 90h	
Size: 8 bit	**Default:** 00h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07:05	Reserved	Reserved				O
04	AEL	Access Error Log — This bit reflects the value of the Hardware Sequencing Status AEL register.				RO
03	FCERR	Flash Cycle Error — Hardware sets this bit to 1 when a programmed access is blocked from running on the SPI interface due to one of the protection policies or when any of the programmed cycle registers is written while a programmed access is already in progress. This bit remains asserted until cleared by software writing a 1 or hardware reset due to a global reset or host partition reset in an Intel® ME enabled system.				RWC
02	CDS	Cycle Done Status — The PCH sets this bit to 1 when the SPI Cycle completes (for example, SCIP bit is 0) after software sets the GO bit. This bit remains asserted until cleared by software writing a 1 or hardware reset due to a global reset or host partition reset in an Intel® ME enabled system. When this bit is set and the SPI SMI# Enable bit is set, an internal signal is asserted to the SMI# generation block. Software must make sure this bit is cleared prior to enabling the SPI SMI# assertion for a new programmed access.				RWC
01	Reserved	Reserved				O
00	SCIP	SPI Cycle In Progress — Hardware sets this bit when software sets the SPI Cycle Go bit in the Command register. This bit remains set until the cycle completes on the SPI interface. Hardware automatically sets and clears this bit so that software can determine when read data is valid and/or when it is safe to begin programming the next command. Software must only program the next command when this bit is 0.				RO

14.1.1.19 Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers)

Table 14-19. Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 91h **Offset End:** 93h	
Size: 24 bit	**Default:** F80000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
23:19	Reserved	Reserved - BIOS must set this field to '11111'b				O
18:16	SCF	SPI Cycle Frequency — This register sets frequency to use for all SPI software sequencing cycles (write, erase, fast read, read status, etc.) except for the read cycle which always run at 20 MHz. 000 = 20 MHz 001 = 33 MHz 100 = 50 MHz All other values reserved. This register is locked when the SPI Configuration Lock-Down bit is set.				RW
15	SME	SPI SMI# Enable — When set to 1, the SPI asserts an SMI# request whenever the Cycle Done Status bit is 1.				RW
14	DS	Data Cycle — When set to 1, there is data that corresponds to this transaction. When 0, no data is delivered for this cycle, and the DBC and data fields themselves are don't cares				RW
13:08	DBC	Data Byte Count — This field specifies the number of bytes to shift in or out during the data portion of the SPI cycle. The valid settings (in decimal) are any value from 0 to 63. The number of bytes transferred is the value of this field plus 1. When this field is 00_0000b, then there is 1 byte to transfer and that 11_1111b means there are 64 bytes to transfer.				RW
07	Reserved	Reserved				O
06:04	COP	Cycle Opcode Pointer — This field selects one of the programmed opcodes in the Opcode Menu to be used as the SPI Command/Opcode. In the case of an Atomic Cycle Sequence, this determines the second command. — R/W.				RW
03	SPOP	Sequence Prefix Opcode Pointer — This field selects one of the two programmed prefix opcodes for use when performing an Atomic Cycle Sequence. A value of 0 points to the opcode in the least significant byte of the Prefix Opcodes register. By making this programmable, the PCH supports flash devices that have different opcodes for enabling writes to the data space vs. status register.				RW

Table 14-19. Offset 91h: Software Sequencing Flash Control Register (SPI Memory Mapped Configuration Registers) (Sheet 2 of 2)

Description:						
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31 :F0	**Offset Start:** 91h **Offset End:** 93h	
Size: 24 bit		**Default:** F80000h			**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	02	ACS	Atomic Cycle Sequence — When set to 1 along with the SCGO assertion, the PCH will execute a sequence of commands on the SPI interface without allowing the LAN component to arbitrate and interleave cycles. The sequence is composed of: Atomic Sequence Prefix Command (8-bit opcode only) Primary Command specified below by software (can include address and data) Polling the Flash Status Register (opcode 05h) until bit 0 becomes 0b. The SPI Cycle in Progress bit remains set and the Cycle Done Status bit remains unset until the Busy bit in the Flash Status Register returns 0.			RW
	01	SCGO	SPI Cycle Go — This bit always returns 0 on reads. However, a write to this register with a 1 in this bit starts the SPI cycle defined by the other bits of this register. The “SPI Cycle in Progress” (SCIP) bit gets set by this action. Hardware must ignore writes to this bit while the Cycle In Progress bit is set. Hardware allows other bits in this register to be programmed for the same transaction when writing this bit to 1. This saves an additional memory write.			RWS
	00	Reserved	Reserved			O

14.1.1.20 Offset 94h: Prefix Opcode Configuration Register (SPI Memory Mapped Configuration Registers)

This register is not writable when the Flash Configuration Lock-Down bit (SPIBAR + 04h:15) is set.

Table 14-20. Offset 94h: Prefix Opcode Configuration Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 94h **Offset End:** 95h	
Size: 16 bit	**Default:** 0000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
15:08	PO1	Prefix Opcode 1— Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.				RW
07:00	PO0	Prefix Opcode 0 — Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.				RW

14.1.1.21 Offset 96h: Opcode Type Configuration Register (SPI Memory Mapped Configuration Registers)

Entries in this register correspond to the entries in the Opcode Menu Configuration register.

The definition below only provides write protection for opcodes that have addresses associated with them. Therefore, any erase or write opcodes that do not use an address should be avoided (for example, "Chip Erase" and "Auto-Address Increment Byte Program")

This register is not writable when the SPI Configuration Lock-Down bit (SPIBAR + 00h:15) is set.

Table 14-21. Offset 96h: Opcode Type Configuration Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31 :F0		**Offset Start:** 96h **Offset End:** 97h	
Size: 16 bit	**Default:** 0000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
15:14	OT7	Opcode Type 7 — See the description for bits 1:0				RW
13:12	OT6	Opcode Type 6 — See the description for bits 1:0				RW
11:10	OT5	Opcode Type 5 — See the description for bits 1:0				RW
09:08	OT4	Opcode Type 4 — See the description for bits 1:0				RW
07:06	OT3	Opcode Type 3 — See the description for bits 1:0				RW
05:04	OT2	Opcode Type 2 — See the description for bits 1:0				RW
03:02	OT1	Opcode Type 1 — See the description for bits 1:0				RW
01:00	OT0	Opcode Type 0 — This field specifies information about the corresponding Opcode 0. This information allows the hardware to 1) know whether to use the address field and 2) provide BIOS and Shared Flash protection capabilities. The encoding of the two bits is: 00 = No address associated with this Opcode; Read cycle type 01 = No address associated with this Opcode; Write cycle type 10 = Address required; Read cycle type 11 = Address required; Write cycle type				RW

14.1.1.22 Offset 98h: Opcode Menu Configuration Register (SPI Memory Mapped Configuration Registers)

Eight entries are available in this register to give BIOS a sufficient set of commands for communicating with the flash device, while also restricting what malicious software can do. This keeps the hardware flexible enough to operate with a wide variety of SPI devices.

It is recommended that BIOS avoid programming Write Enable opcodes in this menu. Malicious software could then perform writes and erases to the SPI flash without using the atomic cycle mechanism. This could cause functional failures in a shared flash environment. Write Enable opcodes should only be programmed in the Prefix Opcodes.

This register is not writable when the SPI Configuration Lock-Down bit (SPIBAR + 00h:15) is set.

Table 14-22. Offset 98h: Opcode Menu Configuration Register (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** 98h **Offset End:** 9Fh	
Size: 64 bit		**Default:** 0000000000000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
63	:56	AO7	Allowable Opcode 7 — See the description for bits 7:0				RW
55	:48	AO6	Allowable Opcode 6 — See the description for bits 7:0				RW
47	:40	AO5	Allowable Opcode 5 — See the description for bits 7:0				RW
39	:32	AO4	Allowable Opcode 4 — See the description for bits 7:0				RW
31	:24	AO3	Allowable Opcode 3 — See the description for bits 7:0				RW
23	:16	AO2	Allowable Opcode 2 — See the description for bits 7:0				RW
15	:08	AO1	Allowable Opcode 1 — See the description for bits 7:0				RW
07	:00	AO0	Allowable Opcode 0 — Software programs an SPI opcode into this field for use when initiating SPI commands through the Control Register.				RW

14.1.1.23 Offset A0h: BIOS Base Address Configuration Register (SPI Memory Mapped Configuration Registers)

Eight entries are available in this register to give BIOS a sufficient set of commands for communicating with the flash device, while also restricting what malicious software can do. This keeps the hardware flexible enough to operate with a wide variety of SPI devices.

Table 14-23. Offset A0h: BIOS Base Address Configuration Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:24	Reserved	Reserved				
23:08	BOSF	Bottom of System Flash— This field determines the bottom of the System BIOS. The PCH will not run programmed commands nor memory reads whose address field is less than this value. this field corresponds to bits 23:8 of the 3-byte address; bits 7:0 are assumed to be 00h for this vector when comparing to a potential SPI address. Note: The SPI host controller prevents any programmed cycle using the address register with an address less than the value in this register. Some flash devices specify that the Read ID command must have an address of 0000h or 0001h. If this command must be supported with these device, it must be performed with the BIOS BAR				RW
07:00	Reserved	Reserved				O

14.1.1.24 Offset B0h: Flash Descriptor Observability Control Register (SPI Memory Mapped Configuration Registers)

This register that can be used to observe the contents of the Flash Descriptor that is stored in the PCH Flash Controller. This register is only applicable when SPI device is in descriptor mode.

Table 14-24. Offset B0h: Flash Descriptor Observability Control Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** B0h **Offset End:** B3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:15	Reserved	Reserved				O
14:12	FDSS	Flash Descriptor Section Select — Selects which section within the loaded Flash Descriptor to observe. 000 = Flash Signature and Descriptor Map 001 = Component 010 = Region 011 = Master 111 = Reserved				RW
11:02	FDSI	Flash Descriptor Section Index — Selects the DW offset within the Flash Descriptor Section to observe.				RW
01:00	Reserved	Reserved				O

14.1.1.25 Offset B4h: Flash Descriptor Observability Data Register (SPI Memory Mapped Configuration Registers)

This register that can be used to observe the contents of the Flash Descriptor that is stored in the PCH Flash Controller.

Table 14-25. Offset B4h: Flash Descriptor Observability Data Register SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** B4h **Offset End:** B7h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:00	FDSD	Flash Descriptor Section Data — Returns the DW of data to observe as selected in the Flash Descriptor Observability Control.				RO

14.1.1.26 Offset C0h: Additional Flash Control Register (SPI Memory Mapped Configuration Registers)

Table 14-26. Offset C0h: Additional Flash Control Register (SPI Memory Mapped Configuration Registers)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:03	Reserved	Reserved				O
02:01	IDCGE	Flash Controller Interface Dynamic Clock Gating Enable: 0 = Flash Controller Interface Dynamic Clock Gating is Disabled 1 = Flash Controller Interface Dynamic Clock Gating is Enabled Other configurations are Reserved.				RW
00	CDCGE	Flash Controller Core Dynamic Clock Gating Enable: 0 = Flash Controller Core Dynamic Clock Gating is Disabled 1 = Flash Controller Core Dynamic Clock Gating is Enabled				RW

14.1.1.27 Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers)

All attributes described in LVSCC must apply to all flash space below the FPBA, even if it spans between two separate flash parts. This register is only applicable when the SPI device is in descriptor mode.

Table 14-27. Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) (Sheet 1 of 2)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:24	Reserved	Reserved				O
	23	LVCL	Vendor Component Lock — This register locks itself when set. 0 = The lock bit is not set 1 = The Vendor Component Lock bit is set. This bit applies to both UVSCC and LVSCC registers.				RW
22	:16	Reserved	Reserved				OO
15	:08	LEO	Lower Erase Opcode — This register is programmed with the Flash erase instruction opcode required by the vendor's Flash component. This register is locked by the Vendor Component Lock (LVCL) bit.				RW
07	:05	Reserved	Reserved				O
	04	LWEWS	Write Enable on Write Status — This register is locked by the Vendor Component Lock (LVCL) bit. 0 = No automatic write of 00h will be made to the SPI flash's status register. 1 = A write of 00h to the SPI flash's status register will be sent on EVERY write and erase to the SPI flash. 06h 01h 00h is the opcode sequence used to unlock the Status register. 1 - This bit should not be set to '1' if there are non volatile bits in the SPI flash's status register. This may lead to premature flash wear out 2 - This is not an atomic sequence. If the SPI component's status register is non-volatile, then BIOS should issue an atomic software sequence cycle to unlock the flash part. 3 - Bit 3 and bit 4 should NOT be both set to '1'.				RW

Table 14-27. Offset C4h: Host Lower Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	03	LWSR	Lower Write Status Required — This register is locked by the Vendor Component Lock (LVCL) bit. 0 = No automatic write of 00h will be made to the SPI flash's status register) 1 = A write of 00h to the SPI flash's status register will be sent on EVERY write and erase to the SPI flash. 50h 01h 00h is the opcode sequence used to unlock the Status register. 1 - This bit should not be set to '1' if there are non volatile bits in the SPI flash's status register. This may lead to premature flash wear out 2 - This is not an atomic sequence. If the SPI component's status register is non-volatile, then BIOS should issue an atomic software sequence cycle to unlock the flash part. 3 - Bit 3 and bit 4 should NOT be both set to '1'.			RW
	02	LWG	Lower Write Granularity — This register is locked by the Vendor Component Lock (LVCL) bit. 0 = 1 Byte 1 = 64 Byte 1 - If more than one Flash component exists, this field must be set to the lowest common write granularity of the different Flash components. 2 - If using 64 B write, BIOS must ensure that multiple byte writes do not occur over 256 B boundaries. This will lead to corruption as the write will wrap around the page boundary on the SPI flash part. This is a a feature page writable SPI flash.			RW
01	:00	LBES	Lower Block/Sector Erase Size — This field identifies the erasable sector size for all Flash components. 00 = 256 Byte 01 = 4 KB 10 = 8 KB 11 = 64 KB This register is locked by the Vendor Component Lock (LVCL) bit. Hardware takes no action based on the value of this register. The contents of this register are to be used only by software and can be read in the HSFSTS.BERASE register in both the BIOS and the GbE program registers if FLA is less than FPBA.			RW

14.1.1.28 Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers)

All attributes described in UVSCC must apply to all flash space equal to or above the FPBA, even if it spans between two separate flash parts. This register is only applicable when SPI device is in descriptor mode.

To prevent this register from being modified you must use LVSCC.VCL bit.

Table 14-28. Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31:16	Reserved	Reserved.				O
15:08	UEO	Upper Erase Opcode — This register is programmed with the Flash erase instruction opcode required by the vendor's Flash component. This register is locked by the Vendor Component Lock (UVCL) bit.				RW
07:05	Reserved	Reserved				O
04	UWEWS	Write Enable on Write Status — This register is locked by the Vendor Component Lock (UVCL) bit. 0 = No automatic write of 00h will be made to the SPI flash's status register) 1 = A write of 00h to the SPI flash's status register will be sent on EVERY write and erase to the SPI flash. 06h 01h 00h is the opcode sequence used to unlock the Status register. 1 - This bit should not be set to '1' if there are non volatile bits in the SPI flash's status register. This may lead to premature flash wear out 2 - This is not an atomic sequence. If the SPI component's status register is non-volatile, then BIOS should issue an atomic software sequence cycle to unlock the flash part. 3 - Bit 3 and bit 4 should NOT be both set to '1'.				RW

Table 14-28. Offset C8h: Host Upper Vendor Specific Component Capabilities Register (SPI Memory Mapped Configuration Registers) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
	03	UWSR	Upper Write Status Required — This register is locked by the Vendor Component Lock (UVCL) bit. 0 = No automatic write of 00h will be made to the SPI flash's status register) 1 = A write of 00h to the SPI flash's status register will be sent on EVERY write and erase to the SPI flash. 50h 01h 00h is the opcode sequence used to unlock the Status register. ***Notes:*** 1. This bit should not be set to '1' if there are non volatile bits in the SPI flash's status register. This may lead to premature flash wear out 2. This is not an atomic sequence. If the SPI component's status register is non-volatile, then BIOS should issue an atomic software sequence cycle to unlock the flash part. 3. Bit 3 and bit 4 should NOT be both set to '1'.			RW
	02	UWG	Upper Write Granularity — This register is locked by the Vendor Component Lock (UVCL) bit. 0 = 1 Byte 1 = 64 Byte 1 - If more than one Flash component exists, this field must be set to the lowest common write granularity of the different Flash components. 2 - If using 64 B write, BIOS must ensure that multiple byte writes do not occur over 256 B boundaries. This will lead to corruption as the write will wrap around the page boundary on the SPI flash part. This is a a feature page writable SPI flash.			RW
01	:00	UBES	Upper Block/Sector Erase Size — This field identifies the erasable sector size for all Flash components. Valid Bit Settings: 00 = 256 Byte 01 = 4 KB 10 = 8 KB 11 = 64 KB This register is locked by the Vendor Component Lock (UVCL) bit. Hardware takes no action based on the value of this register. The contents of this register are to be used only by software and can be read in the HSFSTS.BERASE register in both the BIOS and the GbE program registers if FLA is greater or equal to FPBA.			RW

14.1.1.29 Offset D0h: Flash Partition Boundary (SPI Memory Mapped Configuration Registers)

Table 14-29. Offset D0h: Flash Partition Boundary (SPI Memory Mapped Configuration Registers)

Description:							
View: PCI		**BAR:** SPIBAR		**Bus:Device:Function:** B0:D31:F0		**Offset Start:** D0h **Offset End:** D3h	
Size: 32 bit		**Default:** 00000000h				**Power Well:**	
Bit Range		**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	:13	Reserved	Reserved				O
12	:00	FPBA	Flash Partition Boundary Address — This register reflects the value of Flash Descriptor Component FPBA field.				RO

14.2 Flash Descriptor Records

The following section describes the data structure of the Flash Descriptor on the SPI device. These are not registers within the PCH.

14.2.1 OEM Section

Memory Address: F00h
Default Value: X
Size: 256 Bytes

256 bytes are reserved at the top of the Flash Descriptor for use by the OEM. The information stored by the OEM can only be written during the manufacturing process as the Flash Descriptor read/write permissions must be set to Read Only when the computer leaves the manufacturing floor. The PCH Flash controller does not read this information. FFh is suggested to reduce programming time.

§ §

15.0 UART / WDT (SIW)

15.1 Overview

The Serial I/O unit and Watchdog Timer (SIW) are similar to currently available Super I/O controllers. It is specifically designed for integration into the ILB. It is connected via the LPC bus and currently consists of two UARTs, a Serial Interrupt Controller, a Watchdog Timer and the LPC interface.

15.2 LPC Logical Devices 4 and 5: Serial Ports (UART1 and UART2)

The UARTs are controlled via programmed I/O. The basic programming model is the same for both UARTs with the only difference being the Logical Device Number assigned to each.

15.2.1 UART Register Details

There are 12 registers in the UART. These registers share eight address locations in the I/O address space. Table 15-4 shows the registers and their addresses as offsets of a base address. The state of the Divisor Latch Bit (DLAB), which is the MOST significant bit of the Serial Line Control Register, affects the selection of certain of the UART registers. The DLAB bit must be set high by the system software to access the Baud Rate Generator Divisor Latches.

Table 15-1. Summary of UART Registers in I/O Space (DLAB=0)

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: RBR—Receive Buffer Register" on page 724	00h
00h	00h	"Offset 00h: THR—Transmit Holding Register" on page 724	00h
01h	01h	"Offset 01h: IER—Interrupt Enable Register" on page 725	00h

Table 15-2. Summary of UART Registers in I/O Space (DLAB=1)

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: DLL—Programmable Baud Rate Generator Divisor Latch Register Low" on page 737	02h
01h	01h	"Offset 01h: DLH—Programmable Baud Rate Generator Divisor Latch Register High" on page 738	00h

Table 15-3. Summary of UART Timer Registers in I/O Space

Offset Start	Offset End	Register ID - Description	Default Value
02h	02h	"Offset 02h: IIR—Interrupt Identification Register" on page 726	01h
02h	02h	"Offset 02h: FCR—FIFO Control Register" on page 728	00h
03h	03h	"Offset 03h: LCR—Line Control Register" on page 729	00h
04h	04h	"Offset 04h: MCR—Modem Control Register" on page 731	00h
05h	05h	"Offset 05h: LSR—Line Status Register" on page 733	60h
06h	06h	"Offset 06h: MSR—Modem Status Register" on page 736	00h
07h	07h	"Offset 07h: SCR—Scratchpad Register" on page 737	00h

Table 15-4. Internal Register Descriptions

UART Register Addresses (Base + offset)	DLAB Bit Value	Register Accessed
Base	0	Receive BUFFER (Read-Only)
Base	0	Transmit BUFFER (Write-Only)
Base + 01H	0	Interrupt Enable (Read/Write)
Base + 02H	X	Interrupt I.D. (Read-Only)
Base + 02H	X	FIFO Control (Write-Only)
Base + 03H	X	Line Control (Read/Write)
Base + 04H	X	Modem Control (Read/Write)
Base + 05H	X	Line Status (Read-Only)
Base + 06H	X	Modem Status (Read-Only)
Base + 07H	X	Scratch Pad (Read/Write)
Base	1	Divisor Latch (Lower Byte, Read/Write)
Base + 01H	1	Divisor Latch (Upper Byte, Read/Write)

Note: Base Address for the UART registers listed in Table 15-4 is configurable. See Section 15.4.1, "SIW Configuration Register Summary" on page 746 for details.

15.2.1.1 Offset 00h: RBR—Receive Buffer Register

In non-FIFO mode, this register holds the character received by the UART's Receive Shift Register. If fewer than eight bits are received, the bits are right-justified and the leading bits are zeroed. Reading the register empties the register and resets the Data Ready (DR) bit in the Line Status Register to zero. Other (error) bits in the Line Status Register are not cleared. In FIFO mode, this register latches the value of the data byte at the top of the FIFO.

Table 15-5. Offset 00h: RBR—Receive Buffer Register

Description:					
View: IA F	**Base Address:** Base (IO) (DLAB = 0)			**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RB_7_0	Data byte received (bits [7:0]), least significant bit first		00h	RO

15.2.1.2 Offset 00h: THR—Transmit Holding Register

This register holds the next data byte to be transmitted. When the Transmit Shift Register becomes empty, the contents of the Transmit Holding Register are loaded into the shift register and the transmit data request (TDRQ) bit in the Line Status Register is set to one.

Table 15-6. Offset 00h: THR—Transmit Holding Register

Description:					
View: IA F	**Base Address:** Base (IO) (DLAB = 0)			**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TB_7_0	Data byte received (bits [7:0]), least significant bit first		00h	WO

In FIFO mode, writing to THR puts data to the top of the FIFO. The data at the bottom of the FIFO is loaded to the shift register when it is empty.

15.2.1.3 Offset 01h: IER—Interrupt Enable Register

This register enables five types of interrupts which independently activate the int signal and set a value in the Interrupt Identification Register. Each of the five interrupt types can be disabled by resetting the appropriate bit of the IER register. Similarly, by setting the appropriate bits, selected interrupts can be enabled. Receiver time out interrupt can be configured to be separated from the receive data available interrupt (using the bit 5: COMP)

The use of bit 5 to bit 4 is different from the register definition of standard 16550.

Table 15-7. Offset 01h: IER—Interrupt Enable Register

Description:					
View: IA F	**Base Address:** Base (IO) (DLAB = 0)			**Offset Start:** 01h **Offset End:** 01h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	RSVD	RSVD = 0		00h	RO
05	COMP	**Compatibility Enable:** 0 = Bit 0 of this register also controls RTOIE and bit 4 is RSVD. 1 = Bit 4 of this register controls RTOIE.		0h	RW
04	RTOIE	**Receiver Time Out Interrupt Enable:** 0 = Receiver data Time out interrupt disabled 1 = Receiver data Time out interrupt enabled		0h	RW
03	MIE	**Modem Interrupt Enable:** 0 = Modem Status interrupt disabled 1 = Modem Status interrupt enabled		0h	RW
02	RLSE	**Receiver Line Status Interrupt Enable:** 0 = Receiver Line Status interrupt disabled 1 = Receiver Line Status interrupt enabled		0h	RW
01	TIE	**Transmit Data request Interrupt Enable:** 0 = Transmit FIFO Data Request interrupt disabled 1 = Transmit FIFO Data Request interrupt enabled		0h	RW
00	RAVIE	**Receiver Data Available Interrupt Enable**: When BIT 5 = 1: 0 = Receiver Data Available (Trigger level reached) interrupt disabled 1 = Receiver Data Available (Trigger level reached) interrupt enabled When BIT 5 = 0: 0 = Receiver data Time Out Interrupt also disabled 1 = Receiver data Time Out Interrupt enabled		0h	RW

15.2.1.4 Offset 02h: IIR—Interrupt Identification Register

In order to minimize software overhead during data character transfers, the UART prioritizes interrupts into four levels (listed in Table 15-8) and records these in the Interrupt Identification Register. The Interrupt Identification Register (IIR) stores information indicating that a prioritized interrupt is pending and the source of that interrupt.

Table 15-8. Interrupt Conditions

Priority Level	Interrupt Origin
1 (highest)	Receiver Line Status. One or more error bits were set.
2	Received Data is available. In FIFO mode, trigger level was reached; in non-FIFO mode, RBR has data.
2	Receiver Time out occurred. It happens in FIFO mode only, when there is data in the receive FIFO but no activity for a time period.
3	Transmitter requests data. In FIFO mode, the transmit FIFO is half or more than half empty; in non-FIFO mode, THR is read already.
4	Modem Status: one or more of the modem input signals has changed state

Table 15-9. Offset 02h: IIR—Interrupt Identification Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	FIFOES_1_0	**FIFO Mode Enable Status (bits [1:0]):** 00 Non-FIFO mode is selected. 01 Reserved 10 Reserved 11 FIFO mode is selected (TRFIFOE = 1).		00b	RO
05 :04	Reserved	Reserved		00b	
03	TOD_IID3	**Time Out Detected:** 0 = No time out interrupt is pending. 1 = Time out interrupt is pending. (FIFO mode only)		0b	RO
02 :01	IID_2_1	**Interrupt Source Encoded (bits[2:1]):** 00 Modem Status (CTS, DSR, RI, DCD modem signals changed state) 01 Transmit FIFO requests data 10 Received Data Available 11 Receive error (Overrun, parity, framing, break, FIFO error)		00b	RO
00	P_N	**Interrupt Pending:** 0 = Interrupt is pending. (Active low) 1 = No interrupt is pending.		1b	RO

Table 15-10. Interrupt Identification Register Decode

Interrupt ID bits				Interrupt SET/RESET Function			
3	2	1	0	Priority	Type	Source	RESET Control
0	0	0	1	-	None	No Interrupt is pending.	-
0	1	1	0	Highest	Receiver Line Status	Overrun Error, Parity Error, Framing Error, Break Interrupt.	Reading the Line Status Register.
0	1	0	0	Second Highest	Received Data Available.	Non-FIFO mode: Receive Buffer is full.	Non-FIFO mode: Reading the Receiver Buffer Register.
						FIFO mode: Trigger level was reached.	FIFO mode: Reading bytes until Receiver FIFO drops below trigger level or setting RESETRF bit in FCR register.
1	1	0	0	Second Highest	Character Timeout indication.	FIFO Mode only: At least 1 character is in receiver FIFO and there was no activity for a time period.	Reading the Receiver FIFO or setting RESETRF bit in FCR register.
0	0	1	0	Third Highest	Transmit FIFO Data Request	Non-FIFO mode: Transmit Holding Register Empty	Reading the IIR Register (if the source of the interrupt) or writing into the Transmit Holding Register.
						FIFO mode: Transmit FIFO has half or less than half data.	Reading the IIR Register (if the source of the interrupt) or writing to the Transmitter FIFO.
0	0	0	0	Fourth Highest	Modem Status	Clear to Send, Data Set Ready, Ring Indicator, Received Line Signal Detect	Reading the modem status register

15.2.1.5 Offset 02h: FCR—FIFO Control Register

FCR is a write-only register that is located at the same address as the IIR (IIR is a read-only register). FCR enables/disables the transmitter/receiver FIFOs, clears the transmitter/receiver FIFOs, and sets the receiver FIFO trigger level.

Table 15-11. Offset 02h: FCR—FIFO Control Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	ITL_1_0	**Interrupt Trigger Level (bits [1:0]):** When the number of bytes in the receiver FIFO equals the interrupt trigger level programmed into this field and the Received Data Available Interrupt is enabled (via IER), an interrupt is generated and appropriate bits are set in the IIR. 00 1 byte or more in FIFO causes interrupt 01 RSVD 10 8 bytes or more in FIFO causes interrupt 11 RSVD		00b	WO
05 :03	Reserved	Reserved. Must be programmed to 0.		000b	
02	RESETTF	**Reset transmitter FIFO:** When RESETTF is set to 1, the transmitter FIFO counter logic is set to 0, effectively clearing all the bytes in the FIFO. The TDRQ bit in LSR are set and IIR shows a transmitter requests data interrupt if the TIE bit in the IER register is set. The transmitter shift register is not cleared; it completes the current transmission. After the FIFO is cleared, RESETTF is automatically reset to 0. 0 = Writing 0 has no effect 1 = The transmitter FIFO is cleared (FIFO counter set to 0). After clearing, bit is automatically reset to 0		0b	WO
01	RESETRF	**Reset Receiver FIFO:** When RESETRF is set to 1, the receiver FIFO counter is reset to 0, effectively clearing all the bytes in the FIFO. The DR bit in LSR is reset to 0. All the error bits in the FIFO and the FIFOE bit in LSR are cleared. Any error bits, OE, PE, FE or BI, that had been set in LSR are still set. The receiver shift register is not cleared. If IIR had been set to Received Data Available, it is cleared. After the FIFO is cleared, RESETRF is automatically reset to 0. 0 = Writing 0 has no effect 1 = The receiver FIFO is cleared (FIFO counter set to 0). After clearing, bit is automatically reset to 0		0b	WO
00	TRFIFOE	**Transmit and Receive FIFO Enable:** TRFIFOE enables/disables the transmitter and receiver FIFOs. When TRFIFOE = 1, both FIFOs are enabled (FIFO Mode). When TRFIFOE = 0, the FIFOs are both disabled (non-FIFO Mode). Writing a 0 to this bit clears all bytes in both FIFOs. When changing from FIFO mode to non-FIFO mode and vice versa, data is automatically cleared from the FIFOs. This bit must be 1 when other bits in this register are written or the other bits are not programmed. 0 = FIFOs are disabled 1 = FIFOs are enabled		0b	WO

15.2.1.6 Offset 03h: LCR—Line Control Register

In the Line Control Register (LCR), the system programmer specifies the format of the asynchronous data communications exchange. The serial data format consists of a start bit (logic 0), five to eight data bits, an optional parity bit, and one or two stop bits (logic 1). The LCR has bits for accessing the Divisor Latch and causing a break condition. The programmer can also read the contents of the Line Control Register. The read capability simplifies system programming and eliminates the need for separate storage in system memory.

Table 15-12. Offset 03h: LCR—Line Control Register (Sheet 1 of 2)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 03h **Offset End:** 03h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	DLAB	**Divisor register access bit:** This bit is the Divisor Latch Access Bit. It must be set high (logic 1) to access the Divisor Latches of the Baud Rate Generator during a READ or WRITE operation. It must be set low (logic 0) to access the Receiver Buffer, the Transmit Holding Register, or the Interrupt Enable Register. 0 = Access Transmit Holding register (THR), Receive Buffer Register (RBR) and Interrupt Enable Register. 1 = Access Divisor Latch Registers (DLL and DLH).		0b	RW
06	SB	**Set break:** This bit is the set break control bit. It causes a break condition to be transmitted to the receiving UART. When SB is set to a logic 1, the serial output (TXD) is forced to the spacing (logic 0) state and remains there until SB is set to a logic 0. This bit acts only on the TXD pin and has no effect on the transmitter logic. This feature enables the processor to alert a terminal in a computer communications system. If the following sequence is executed, no erroneous characters are transmitted because of the break: Load 00H in the Transmit Holding register in response to a TDRQ interrupt After TDRQ goes high (indicating that 00H is being shifted out), set the break bit before the parity or stop bits reach the TXD pin Wait for the transmitter to be idle (TEMT = 1) and clear the break bit when normal transmission has to be restored During the break, the transmitter can be used as a character timer to accurately establish the break duration. In FIFO mode, wait for the transmitter to be idle (TEMT=1) to set and clear the break bit. 0 = No effect on TXD output 1 = Forces TXD output to 0 (space)		0b	RW

Table 15-12. Offset 03h: LCR—Line Control Register (Sheet 2 of 2)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 03h **Offset End:** 03h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05	STKYP	**Sticky Parity:** This bit is the "sticky parity" bit, which can be used in multiprocessor communications. When PEN and STKYP are logic 1, the bit that is transmitted in the parity bit location (the bit just before the stop bit) is the complement of the EPS bit. If EPS is 0, then the bit at the parity bit location is transmitted as a 1. In the receiver, if STKYP and PEN are 1, then the receiver compares the bit that is received in the parity bit location with the complement of the EPS bit. If the values being compared are not equal, the receiver sets the Parity Error bit in LSR and causes an error interrupt if line status interrupts were enabled. For example, if EPS is 0, the receiver expects the bit received at the parity bit location to be 1. If it is not, then the parity error bit is set. By forcing the bit value at the parity bit location, rather than calculating a parity value, a system with a master transmitter and multiple receivers can identify some transmitted characters as receiver addresses and the rest of the characters as data. If PEN = 0, STKYP is ignored. 0 = No effect on parity bit 1 = Forces parity bit to be opposite of EPS bit value		0b	RW
04	EPS	**Even parity Select:** This bit is the even parity select bit. When PEN is a logic 1 and EPS is a logic 0, an odd number of logic ones is transmitted or checked in the data word bits and the parity bit. When PEN is a logic 1 and EPS is a logic 1, an even number of logic ones is transmitted or checked in the data word bits and parity bit. If PEN = 0, EPS is ignored. 0 = Sends or checks for odd parity 1 = Sends or checks for even parity		0b	RW
03	PEN	**Parity enable:** This is the parity enable bit. When PEN is a logic 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. (The parity bit is used to produce an even or odd number of ones when the data word bits and the parity bit are summed.) 0 = No parity function 1 = Allows parity generation and checking		0b	RW
02	STB	**Stop bits:** This bit specifies the number of stop bits transmitted and received in each serial character. If STB is a logic 0, one stop bit is generated in the transmitted data. If STB is a logic 1 when a 5-bit word length is selected via bits 0 and 1, then 1 and one half stop bits are generated. If STB is a logic 1 when either a 6, 7, or 8-bit word is selected, then two stop bits are generated. The receiver checks the first stop bit only, regardless of the number of stop bits selected. 0 = 1 stop bit 1 = 2 stop bits, except for 5-bit character then 1-1/2 bits		0b	RW
01 :00	WLS_1_0	**Word Length select:** The Word Length Select bits specify the number of data bits in each transmitted or received serial character. 00 5-bit character (default) 01 6-bit character 10 7-bit character 11 8-bit character		00b	RW

15.2.1.7 Offset 04h: MCR—Modem Control Register

This 8-bit register controls the interface with the modem or data set (or a peripheral device emulating a modem).

Table 15-13. Offset 04h: MCR—Modem Control Register (Sheet 1 of 2)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 04h **Offset End:** 04h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :05	Reserved	Reserved		000b	
04	LOOP	**Loop back test mode:** This bit provides a local Loopback feature for diagnostic testing of the UART. When LOOP is set to a logic 1, the following occurs: The transmitter serial output is set to a logic 1 state. The OUT2# signal is forced to a logic 1 state. The receiver serial input is disconnected from the pin. The output of the Transmitter Shift register is "looped back" into the receiver shift register input. The four modem control inputs (CTS#, DSR#, DCD#, and RI#) are disconnected from the pins and the modem control output pins (RTS# and DTR#) are forced to their inactive state. • Coming out of the loopback test mode may result in unpredictable activation of the delta bits (bits 3:0) in the Modem Status Register (MSR). It is recommended that the MSR be read once to clear the delta bits in the MSR. The lower four bits of the Modem Control register are connected to the upper four Modem Status register bits: • DTR = 1 forces DSR to a 1 • RTS = 1 forces CTS to a 1 • OUT1 = 1 forces RI to a 1 • OUT2= 1 forces DCD to a 1 In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit and receive data paths of the UART. The transmit, receive and modem control interrupts are operational, except the modem control interrupts are activated by Control register bits, not the modem control inputs. A break signal can also be transferred from the transmitter section to the receiver section in loopback mode. 0 = Normal UART operation 1 = Test mode UART operation		0b	RW
03	OUT2	**OUT2# signal control:** This bit controls the OUT2# output. When the OUT2 bit is set, OUT2# is asserted low. When the OUT2 bit is cleared, OUT2# is deasserted (set high). Outside of the UART module, the OUT2# signal is used to connect the UART's interrupt output to the Interrupt Controller unit. 0 = OUT2# signal is 1, which disables the UART interrupt. 1 = OUT2# signal is 0.		0b	RW

Table 15-13. Offset 04h: MCR—Modem Control Register (Sheet 2 of 2)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 04h **Offset End:** 04h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	OUT1	**Test bit:** This bit is used only in Loopback test mode. See (LOOP) Above.		0b	RW
01	RTS	**Request to Send:** This bit controls the Request to Send (RTS#) output pin. Bit 1 affects the RTS# output in a manner identical to that described below for the DTR bit. 0 = RTS# pin is 1 1 = RTS# pin is 0		0b	RW
00	DTR	**Data Terminal Ready:** This bit controls the Data Terminal Ready output. When bit 0 is set to a logic 1, the DTR# output is forced to a logic 0. When bit 0 is reset to a logic 0, the DTR# output pin is forced to a logic 1. • The DTR# output of the UART may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding modem or data set. 0 = DTR# pin is 1 1 = DTR# pin is 0		0b	RW

15.2.1.8 Offset 05h: LSR—Line Status Register

This register provides status information to the processor concerning the data transfers. Bits 5 and 6 show information about the transmitter section. The rest of the bits contain information about the receiver.

In non-FIFO mode, three of the LSR register bits, parity error, framing error, and break interrupt, show the error status of the character that has just been received. In FIFO mode, these three bits of status are stored with each received character in the FIFO. LSR shows the status bits of the character at the top of the FIFO. When the character at the top of the FIFO has errors, the LSR error bits are set and are not cleared until software reads LSR, even if the character in the FIFO is read and a new character is now at the top of the FIFO.

Bits one through four are the error conditions that produce a receiver line status interrupt when any of the corresponding conditions are detected and the interrupt is enabled. These bits are not cleared by reading the erroneous byte from the FIFO or receive buffer. They are cleared only by reading LSR. In FIFO mode, the line status interrupt occurs only when the erroneous byte is at the top of the FIFO. If the erroneous byte being received is not at the top of the FIFO, an interrupt is generated only after the previous bytes are read and the erroneous byte is moved to the top of the FIFO.

Table 15-14. Offset 05h: LSR—Line Status Register (Sheet 1 of 3)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 05h **Offset End:** 05h	
Size: 8 bit	**Default:** 60h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	FIFOE	**FIFO Error Status:** This bit is reset only when all the error bytes have been read from the FIFO. A processor read to the Line Status register does not reset this bit. Non-FIFO mode: 0 = Bit is always "0" indicating no FIFO. FIFO mode: 0 = All error bytes have been read from the FIFO 1 = At least one character in the receiver FIFO contains a parity error, framing error, or break indication. When DMA requests are enabled (IER bit7 is set to 1) and FIFOE is set to 1, no receive DMA request is generated even though the receive FIFO reaches the trigger level and the error interrupt is generated. When DMA requests are not enabled (IER bit7 is set to 0), FIFOE set to 1 does not generate an interrupt.		0b	RO
06	TEMT	**Transmitter Empty:** Non-FIFO mode: 0 = Either the Transmit Holding register or the Tansmitter Shift register contain a data character. 1 = The Transmit Holding register and the Transmitter Shift register are both empty. FIFO mode: 0 = Either the transmitter FIFO or the Transmit Shift register contain a data character. 1 = The transmitter FIFO and the Transmit Shift register are both empty.		1b	RO
05	TRDQ	**Transmit Data Request:** TDRQ indicates that the UART is ready to accept a new character for transmission. In addition, this bit causes the UART to issue an interrupt to the processor when the transmit data request interrupt enable is set high. Non-FIFO mode: 0 = No character transferred from the Transmit Holding register into the Transmit Shift register. 1 = A character has transferred from the Transmit Holding register into the Transmit Shift register. ***Note:*** Bit is reset to logic 0 with the loading of the Transmit Holding register by the processor. FIFO mode: 0 = When at least one byte is written to the transmit FIFO. When more than 16 characters are loaded into the FIFO, the excess characters are lost. 1 = Transmit FIFO is empty or the RESETTF bit in FCR, has been set to 1.		1b	RO

Table 15-14. Offset 05h: LSR—Line Status Register (Sheet 2 of 3)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 05h **Offset End:** 05h	
Size: 8 bit	**Default:** 60h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
04	BI	**Break Interrupt:** BI is set to a logic 1 when the received data input is held in the spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + parity bit + stop bits).The BI is reset to a logic "0" when the processor reads the Line Status register. 0 = No Break signal has been received. 1 = Break signal occurred. In FIFO mode, only one character (equal to 00H), is loaded into the FIFO regardless of the length of the break condition. BI shows the break condition for the character at the top of the FIFO, not the most recently received character.		0h	RO
03	FE	**Framing Error:** FE indicates that the received character did not have a valid stop bit. This bit is reset to a logic "0" when the processor reads the Line Status Register. 0 = No Framing error. 1 = Invalid stop bit has been detected. FE is set to a logic 1 when the bit following the last data bit or parity bit is detected as a logic 0 (spacing level). If the Line Control register had been set for two stop bit mode, the receiver does not check for a valid second stop bit. The FE indicator is reset when the processor reads the Line Status Register. The UART resynchronizes after a framing error. To do this it assumes that the framing error was due to the next start bit, so it samples this "start" bit twice and then takes in the "data". In FIFO mode FE shows a Framing error for the character at the top of the FIFO, not for the most recently received character.		0h	RO
02	PE	**Parity Error:** PE indicates that the received data character does not have the correct even or odd parity, as selected by the even parity select bit. The PE is set to logic 1 upon detection of a parity error and is reset to a logic 0 when the processor reads the Line Status register. In FIFO mode, PE shows a parity error for the character at the top of the FIFO, not the most recently received character. 0 = No Parity error. 1 = Parity error has occurred.		0h	RO

Table 15-14. Offset 05h: LSR—Line Status Register (Sheet 3 of 3)

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 05h **Offset End:** 05h	
Size: 8 bit	**Default:** 60h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
01	OE	**Overrun Error:** In non-FIFO mode, OE indicates that data in the receiver buffer register was not read by the processor before the next character was transferred into the receiver buffer register, thereby destroying the previous character. In FIFO mode, OE indicates that all 16 bytes of the FIFO are full and the most recently received byte has been discarded. The OE indicator is set to a logic "1" upon detection of an overrun condition and reset when the processor reads the Line Status Register. 0 = No data has been lost 1 = Received data has been lost.		0h	RO
00	DR	**Data Ready:** DR is set to logic 1 when a complete incoming character has been received and transferred into the Receiver Buffer Register (RBR) or the FIFO. In non-FIFO mode, DR is reset to 0 when the receive buffer is read. In FIFO mode, DR is reset to a logic 0 if the FIFO is empty (last character has been read from Receiver Buffer Register) or the RESETRF bit is set in FCR. 0 = No data has been received 1 = Data available in RBR or the FIFO.		0h	RO

15.2.1.9 Offset 06h: MSR—Modem Status Register

This 8-bit register provides the current state of the control lines from the modem or data set (or a peripheral device emulating a modem) to the processor. In addition to this current state information, four bits of the Modem Status register provide change information. Bits 03:00 are set to a logic 1 when a control input from the Modem changes state. They are reset to a logic 0 when the processor reads the Modem Status register.

When bits 0, 1, 2, or 3 are set to logic 1, a Modem Status interrupt is generated if bit 3 of the Interrupt Enable Register is set.

Table 15-15. Offset 06h: MSR—Modem Status Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 06h **Offset End:** 06h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	DCD	**Data Carrier Detect:** This bit is the complement of the Data Carrier Detect (DCD#) input. This bit is equivalent to bit OUT2 of the Modem Control register if LOOP in the MCR is set to 1. 0 = DCD# pin is 1 1 = DCD# pin is 0		0b	RO
06	RI	**Ring Indicator:** This bit is the complement of the ring Indicator (RI#) input. This bit is equivalent to bit OUT1 of the Modem Control register if LOOP in the MCR is set to 1. 0 = RI# pin is 1 1 = RI# pin is 0		0b	RO
05	DSR	**Data Set Ready:** This bit is the complement of the Data Set Ready (DSR#) input. This bit is equivalent to bit DTR of the Modem Control register if LOOP in the MCR is set to 1. 0 = DSR# pin is 1 1 = DSR# pin is 0		0b	RO
04	CTS	**Clear to Send:** This bit is the complement of the Clear to Send (CTS#) input. This bit is equivalent to bit RTS of the Modem Control register if LOOP in the MCR is set to 1. 0 = CTS# pin is 1 1 = CTS# pin is 0		0b	RO
03	DDCD	**Delta Data Carrier Detect:** 0 = No change in DCD# pin since last read of the MSR 1 = DCD# pin has changed state		0b	RO
02	TERI	**Trailing Edge Ring Indicator:** 0 = RI# pin has not changed from 0 to 1 since last read of the MSR 1 = RI# pin has changed from 0 to 1		0b	RO
01	DDSR	**Delta Data Set Ready:** 0 = No change in DSR# pin since last read of the MSR 1 = DSR# pin has changed state		0b	RO
00	DCTS	**Delta Clear To Send:** 0 = No change in CTS# pin since last read of the MSR 1 = CTS# pin has changed state		0b	RO

15.2.1.10 Offset 07h: SCR—Scratchpad Register

This 8-bit read/write register has no effect on the UART. It is intended as a scratchpad register for use by the programmer.

Table 15-16. Offset 07h: SCR—Scratchpad Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 07h **Offset End:** 07h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SP_7_0	No effect on UART functionality		00h	RW

15.2.1.11 Offset 00h: DLL—Programmable Baud Rate Generator Divisor Latch Register Low

See Section 4.13.4.3, “Programmable Baud Rate Generator” on page 191.

Table 15-17. Offset 00h: DLL—Programmable Baud Rate Generator Divisor Latch Register Low

Description:					
View: IA F	**Base Address:** Base (IO) (DLAB = 1)			**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 02h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BR_7_0	Low byte compare value to generate baud rate		02h	RW

15.2.1.12 Offset 01h: DLH—Programmable Baud Rate Generator Divisor Latch Register High

See Section 4.13.4.3, "Programmable Baud Rate Generator" on page 191.

Table 15-18. Offset 01h: DLH—Programmable Baud Rate Generator Divisor Latch Register High

Description:					
View: IA F	**Base Address:** Base (IO) (DLAB = 1)			**Offset Start:** 01h **Offset End:** 01h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BR_15_8	High byte compare value to generate baud rate		00h	RW

15.3 Logical Device 6: Watchdog Timer

The Watchdog timer provides a resolution that ranges from 1 µs to 10 minutes. The timer uses a 35-bit down-counter.

The counter is loaded with the value from the first Preload register. The timer is then enabled and it starts counting down. The time at which the WDT first starts counting down is called the first stage. If the host fails to reload the WDT before the 35-bit down counter reaches zero the WDT generates an internal interrupt.

After the interrupt is generated the WDT loads the value from the second Preload register into the WDT's 35-bit Down-Counter and starts counting down. The WDT is now in the second stage. If the host still fails to reload the WDT before the second timeout, the WDT drives the WDT_TOUT# pin low and sets the timeout bit (WDT_TIMEOUT). This bit indicates that the System has become unstable. The WDT_TOUT# pin is held low until the system is Reset or the WDT times out again (Depends on TOUT_CNF). The process of reloading the WDT involves the following sequence of writes:

1. Write "80" to offset BAR1 + 0Ch
2. Write "86" to offset BAR1 + 0Ch
3. Write '1' to WDT_RELOAD in Reload Register.

The same process is used for setting the values in the preload registers. The only difference exists in step 3. Instead of writing a '1' to the WDT_RELOAD, you write the desired preload value into the corresponding Preload register. This value is not loaded into the 35-bit down counter until the next time the WDT reenters the stage. For example, if Preload Value 2 is changed, it is not loaded into the 35-bit down counter until the next time the WDT enters the second stage.

15.3.1 Watchdog Timer Register Details

All registers not mentioned are reserved.

Table 15-19. Summary of Watchdog Timer Registers in I/O Space

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: PV1R0—Preload Value 1 Register 0" on page 739	FFh
01h	01h	"Offset 01h: PV1R1—Preload Value 1 Register 1" on page 740	FFh
02h	02h	"Offset 02h: PV1R2—Preload Value 1 Register 2" on page 740	0Fh
04h	04h	"Offset 04h: PV2R0—Preload Value 2 Register 0" on page 741	FFh
05h	05h	"Offset 05h: PV2R1—Preload Value 2 Register 1" on page 741	FFh
06h	06h	"Offset 06h: PV2R2—Preload Value 2 Register 2" on page 742	0Fh
08h	08h	"Offset 08h: GISR—General Interrupt Status Register" on page 742	00h
0Ch	0Ch	"Offset 0Ch: RR0—Reload Register 0" on page 743	00h
0Dh	0Dh	"Offset 0Dh: RR1—Reload Register 1" on page 743	00h
10h	10h	"Offset 10h: WDTCR—WDT Configuration Register" on page 744	00h
18h	18h	"Offset 18h: WDTLR—WDT Lock Register" on page 745	00h

Note: Base Address for the Watchdog Timer registers, listed in this section, is configurable. See Section 15.4.1, "SIW Configuration Register Summary" on page 746 for details.

15.3.1.1 Offset 00h: PV1R0—Preload Value 1 Register 0

Table 15-20. Offset 00h: PV1R0—Preload Value 1 Register 0

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PLOAD1_7_0	**Preload_Value_1 [7:0]:** This register is used to hold the bits 0 through 7 of the preload value 1 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		FFh	RW

15.3.1.2 Offset 01h: PV1R1—Preload Value 1 Register 1

Table 15-21. Offset 01h: PV1R1—Preload Value 1 Register 1

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 01h **Offset End:** 01h	
Size: 8 bit	**Default:** FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PLOAD1_15_8	**Preload_Value_1 [15:8]:** This register is used to hold the bits 8 through 15 of the preload value 1 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		FFh	RW

15.3.1.3 Offset 02h: PV1R2—Preload Value 1 Register 2

Table 15-22. Offset 02h: PV1R2—Preload Value 1 Register 2

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 0Fh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 04	Reserved	Reserved		0h	
03 : 00	PLOAD_19_16	**Preload_Value_1 [19:16]:** This register is used to hold the bits 16 through 19 of the preload value 1 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the first stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		Fh	RW

15.3.1.4 Offset 04h: PV2R0—Preload Value 2 Register 0

Table 15-23. Offset 04h: PV2R0—Preload Value 2 Register 0

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 04h **Offset End:** 04h	
Size: 8 bit	**Default:** FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PLOAD2_7_0	**Preload_Value_2 [7:0]:** This register is used to hold the bits 0 through 7 of the preload value 2 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the second stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		FFh	RW

15.3.1.5 Offset 05h: PV2R1—Preload Value 2 Register 1

Table 15-24. Offset 05h: PV2R1—Preload Value 2 Register 1

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 05h **Offset End:** 05h	
Size: 8 bit	**Default:** FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PLOAD2_15_8	**Preload_Value_2 [15:8]:** This register is used to hold the bits 8 through 15 of the preload value 2 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the second stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		FFh	RW

15.3.1.6 Offset 06h: PV2R2—Preload Value 2 Register 2

Table 15-25. Offset 06h: PV2R2—Preload Value 2 Register 2

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 06h **Offset End:** 06h	
Size: 8 bit	**Default:** 0Fh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 04	Reserved	Reserved		0h	
03 : 00	PLOAD2_19_16	**Preload_Value_2 [19:16]:** This register is used to hold the bits 16 through 19 of the preload value 2 for the WDT Timer. The Value in the Preload Register is automatically transferred into the 35-bit down counter every time the WDT enters the second stage. The value loaded into the preload register needs to be one less than the intended period. This is because the timer makes use of zero-based counting (for example, zero is counted as part of the decrement).		Fh	RW

15.3.1.7 Offset 08h: GISR—General Interrupt Status Register

Table 15-26. Offset 08h: GISR—General Interrupt Status Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 01	Reserved	Reserved		00h	
00	SERIRQACT	**Watchdog Timer SERIRQ Interrupt Active (1st Stage):** This bit is set when the first Stage of the 35-bit Down Counter Reaches zero. An SERIRQ interrupt is generated if WDT_INT_TYPE is configured to do so (See WDT Configuration Register). This is a sticky bit and is only cleared by writing a '1'. 0 = No Interrupt 1 = Interrupt Active ***Note:*** This bit is not set in free running mode.		0h	RWC

15.3.1.8 Offset 0Ch: RR0—Reload Register 0

Table 15-27. Offset 0Ch: RR0—Reload Register 0

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	Reserved	**Reserved.** Must be programmed to 0.		00h	

15.3.1.9 Offset 0Dh: RR1—Reload Register 1

Table 15-28. Offset 0Dh: RR1—Reload Register 1

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 02	Reserved	Reserved		00h	
01	TOUT	**WDT_TIMEOUT:** This bit is located in the RTC Well and it's value is not lost if the host resets the system. It is set to '1' if the host fails to reset the WDT before the 35-bit Down-Counter reaches zero for the second time in a row. This bit is cleared by performing the Register Unlocking Sequence followed by a '1' to this bit. 0 = Normal (Default) 1 = System has become unstable. ***Note:*** In free running mode this bit is set every time the down counter reaches zero.		0h	RW
00	RELOAD	**WDT_RELOAD:** To prevent a timeout the host must perform the Register Unlocking Sequence followed by a '1' to this bit.		0h	RW

15.3.1.10 Offset 10h: WDTCR—WDT Configuration Register

Table 15-29. Offset 10h: WDTCR—WDT Configuration Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 10h **Offset End:** 10h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 06	Reserved	Reserved		00h	
05	WDT_TOUT_EN	**WDT Timeout Output Enable:** This bit indicates whether or not the WDT toggles the external WDT_TOUT# pin if the WDT times out. 0 = Enabled (Default) 1 = Disabled		0h	RW
04 : 03	Reserved	Reserved		00h	
02	WDT_PRE_SEL	**WDT Prescaler Select:** The WDT provides two options for prescaling the main Down Counter. The preload values are loaded into the main down counter right justified. The prescaler adjusts the starting point of the 35-bit down counter. 0 = The 20-bit Preload Value is loaded into bits 34:15 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^{15}. The approximate clock generated is 1 KHz, (1 ms to 10 min). (Default) 1 = The 20-bit Preload Value is loaded into bits 24:05 of the main down counter. The resulting timer clock is the PCI Clock (33 MHz) divided by 2^{5}. The approximate clock generated is 1 MHz, (1 µs to 1sec)		0h	RW
01 : 00	WDT_INT_TYPE	**WDT_INT_TYPE:** The WDT timer supports programmable routing of interrupts. The set of bits allows the user to choose the type of interrupt desired if the WDT reached the end of the first stage without being reset. The interrupt status is reported in the WDT General Interrupt Status register. 00 SERIRQ (Default) 01 reserved 10 reserved 11 Disabled ***Note:*** SERIRQ is Active Low		00h	RW

15.3.1.11 Offset 18h: WDTLR—WDT Lock Register

Table 15-30. Offset 18h: WDTLR—WDT Lock Register

Description:					
View: IA F	**Base Address:** Base (IO)			**Offset Start:** 18h **Offset End:** 18h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 03	Reserved	Reserved		0h	
02	WDT_TOUT_CNF	**WDT Timeout Configuration:** This register is used to choose the functionality of the timer. 0 = Watchdog Timer Mode: When enabled (for example, WDT_ENABLE goes from '0' to '1') the timer reloads Preload Value 1 and start decrementing. (Default) Upon reaching the second stage timeout the WDT_TOUT# is driven low once and does not change again until Power is cycled or a hard reset occurs. 1 = Free Running Mode: WDT_TOUT# changes from previous state when the next timeout occurs. The timer ignores the first stage. The timer only uses Preload Value 2. In this mode the timer is restarted whenever WDT_ENABLE goes from a 0 to a 1. This means that the timer reloads Preload Value 2 and start decrementing every time it is enabled. In free running mode it is not necessary to reload the timer as it is done automatically every time the descrementer reaches zero.		0h	RW
01	WDT_ENABLE	**Watchdog Timer Enable:** The following bit enables or disables the WDT. 0 = Disabled (Default) 1 = Enabled ***Note:*** This bit cannot be modified if WDT_LOCK has been set. • In free-running mode Preload Value 2 is reloaded into the down counter every time WDT_ENABLE goes from '0' to '1'. • In WDT mode Preload Value 1 is reloaded every time WDT_ENABLE goes from '0' to '1' or the WDT_RELOAD bit is written using the proper sequence of writes (See Register Unlocking Sequence). When the WDT second stage timeout occurs, a reset must happen. • Software must guarantee that a timeout is not about to occur before disabling the timer. A reload sequence is suggested.		0h	RW
00	WDT_LOCK	**Watchdog Timer Lock:** Setting this bit locks the values of this register until a hard-reset occurs or power is cycled. 0 = Unlocked (Default) 1 = Locked ***Note:*** Writing a "0" has no effect on this bit. Write is only allowed from "0" to "1" once. It cannot be changed until either power is cycled or a hard-reset occurs.		0h	RWL

15.4 SIW Configuration

The configuration of the SIW is flexible and is based on the configuration architecture implemented in typical Plug-and-Play components. The SIW is designed for motherboard applications in which the resources required by their components are known. With its flexible resource allocation architecture, the SIW allows the BIOS to assign resources at POST.

15.4.1 SIW Configuration Register Summary

The default values are defined with an h for hex, a bi for binary, or 00 for zero. If no letter follows the default value, assume it is a binary number.

Warning: Address locations that are not listed are considered reserved register locations. Reads to reserved registers may return non-zero values. Writes to reserved locations may cause system failure. Reserved bits are Read Only.

Table 15-31. Configuration Register Summary (Sheet 1 of 2)

Global Configuration Registers			
Index	**Type**	**Default**	**Configuration Register**
07h	RW	00h	Logical Device Number
20h	R	00h	Device ID
21h	R	01h	Device Rev
28h	RW	01h	SIW I/F (wait states)
29h	RW	02h	SIRQ Configuration
2Eh	RW	00h	Test Mode Configuration Register
Logical Device 4 Registers (Serial Port 1)			
30h	RW	00h	Enable
60h	RW	00h	Base I/O Address MSB
61h	RW	00h	Base I/O Address LSB
70h	RW	00h	Primary Interrupt Select
74h	RW	04h	RSVD
75h	RW	04h	RSVD
F0h	RW	00h	RSVD
Logical Device 5 Registers (Serial Port 2)			
30h	RW	00h	Enable
60h	RW	00h	Base I/O Address MSB
61h	RW	00h	Base I/O Address LSB
70h	RW	00h	Primary Interrupt Select
74h	RW	04h	RSVD
75h	RW	04h	RSVD
F0h	RW	00h	RSVD

Table 15-31. Configuration Register Summary (Sheet 2 of 2)

Logical Device 6 Registers (Watchdog Timer)			
30h	RW	00h	Enable
60h	RW	00h	Base I/O Address MSB
61h	RW	00h	Base I/O Address LSB
70h	RW	00h	Primary Interrupt Select

15.4.1.1 Global Control/Configuration Registers [00h - 2Fh]

The chip-level (global) registers lie in the address range [00h-2Fh]. The design MUST use all eight bits of the ADDRESS Port for register selection. All unimplemented registers and bits ignore writes and return zero when read.

The INDEX PORT is used to select a configuration register in the chip. The DATA PORT is then used to access the selected register. These registers are accessible only in the Configuration Mode.

Register	Address (Type)	Description
Logical Device # Default = 00h	07h (RW)	**Logical Device Select**: A write to this register selects the current logical device. This allows access to the control and configuration registers for each logical device.
Device ID Default = 00h	20h (R)	**Device ID:** A read only register which provides the Device ID.
Device Rev Default = 01h	21h (R)	**Device Rev:** A read only register which provides device revision information.
SIW Interface Default = 01h	28h	Bit 7:1 RSVD = 0 Bit 0 LPC bus wait states 0 = Not Supported 1 = Long wait states (sync 6)
SIW Configuration Default = 02h	29h (RW - bit 0, 2, 3) (R - bit 1)	Bit 0 SIRQ enable 1 =enabled; enabled logical devices participate in interrupt generation 0 =disabled; serial interrupts disabled Bit 1 IRQ mode (READ ONLY, WRITES IGNORED) 1 =Continuous mode 0 =Quiet mode Bit 3:2 UART_CLK predivide 00 = divide by 1 01 = divide by 8 10 = divide by 26 11 = reserved Bit 7:4 RSVD = 0

Register	Address (Type)	Description
SIW Monitor Port Control Register Default = 00h	2D (RO - bit 7) (RW - bits 6:0)	Bit 0 UART1 Monitor Port Enable (UART1_MONPORTEN): Setting this bit enables the monitor port for UART1. This signal turns on all 8 UART2 monitor ports. Note: wdt_monporten, uart2_monporten and uart1_monporten must be set in a mutually exclusive manner for example, only one monitor port enable must be set at one time. Bit 1 UART2 Monitor Port Enable (UART2_MONPORTEN): Setting this bit enables the monitor port for UART2. This signal turns on all 8 UART1 monitor ports. Note: wdt_monporten, uart2_monporten and uart1_monporten must be set in a mutually exclusive manner for example, only one monitor port enable must be set at one time. Bit 2 WDT Monitor Port Enable (WDT_MONPORTEN): Setting this bit enables the monitor port for WDT. This signal turns on all 16 SIW monitor ports. Note: wdt_monporten, uart2_monporten and uart1_monporten must be set in a mutually exclusive manner for example, only one monitor port enable must be set at one time. Bit 6:3 Monitor Port Slot Select[3:0] (MONPORTSEL): These bits select which Monitor Port Slot is enabled. These bits are used to select up to 16 slots within each UART1 and UART2 source group. Note: WDT has only 1 slot. So, port slot selection is not required. Bit 7 RSVD = 0
Default = 00h	2Eh	Reserved

15.4.1.2 Logical Device Configuration Registers [30h — FFh]

Used to access the registers that are assigned to each logical unit. This chip supports three logical units and has three sets of logical device registers. The three logical devices are UART1, UART2 and Watchdog Timer. A separate set (bank) of control and configuration registers exists for each logical device and is selected with the Logical Device # Register.

The INDEX PORT is used to select a specific logical device register. These registers are then accessed through the DATA PORT.

The Logical Device registers are accessible only when the device is in the Configuration State. The logical register addresses are shown in the tables below.

The following pseudo code shows how to select and set the LPC Generic I/O Range = to the Watch Dog Timer (WDT) I/O Base Address.

Assume LPC Generic I/O Range#4, size=32B, LPC_Address=6000h

1. CFG WR, BDF= 0,31,0, Offset= 90h = 001C6001

LPC Generic I/O Range#s are located at

Offset= 84h -- LPC Generic I/O Range#1

Offset= 88h -- LPC Generic I/O Range#2

Offset= 8Ch -- LPC Generic I/O Range#3

Offset= 90h -- LPC Generic I/O Range#4

Enable decoding of I/O locations 4Eh and 4Fh to LPC.

1. CFG WR, BDF= 0,31,0, Offset= 82h, bit 13 Microcontroller Enable #2 (ME2):

Enter configuration mode

1. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00800000h
2. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00860000h

Select logical device 6

1. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00070000h
2. I/O WR, addr = 4Fh, data_len = 1, be_first = 8, be_last = 0, data = 06000000h

Set WDT I/O Base Address = selected LPC Generic I/O Rage LPC_Address

1. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00600000h
2. I/O WR, addr = 4Fh, data_len = 1, be_first = 8, be_last = 0, data = 60000000h
3. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00610000h
4. I/O WR, addr = 4Fh, data_len = 1, be_first = 8, be_last = 0, data = 00000000h

Set WDT enable

1. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00300000h
2. I/O WR, addr = 4Fh, data_len = 1, be_first = 8, be_last = 0, data = 01000000h

Exit configuration mode

1. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00680000h
2. I/O WR, addr = 4Eh, data_len = 1, be_first = 4, be_last = 0, data = 00080000h

Table 15-32. Logical Device 4 (Serial Port 1) (Sheet 1 of 2)

Logical Device Register	Address	Description

Enable
Default = 00h
	30h
	(RW)
		Bits[7:1] Reserved, set to zero.
		Bit[0]
		1 = enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive

I/O Base Address Default = 00h	60h (RW) 61h (Bits 7:3 RW Bits 2:0 RO)	Registers 60h (MSB) and 61h (LSB) set the base address for the device. Note: Decode is on 8 Byte boundaries Comm Decode Ranges 3F8 - 3FF (COM 1) 2F8 - 2FF (COM 2) 220 - 227 228 - 22F 238 - 23F 2E8 - 2EF (COM 4) 338 - 33F 3E8 - 3EF (COM 3)

Table 15-32. Logical Device 4 (Serial Port 1) (Sheet 2 of 2)

Logical Device Register	Address	Description
Primary Interrupt Select Default = 00h	70h (RW)	Bits[3:0] selects which interrupt level is used for the primary Interrupt. 00= no interrupt selected 01= IRQ1 02= IRQ2 03= IRQ3 04= IRQ4 05= IRQ5 06= IRQ6 07= IRQ7 08= IRQ8 09= IRQ9 0A= IRQ10 0B= IRQ11 0C= IRQ12 0D= IRQ13 0E= IRQ14 0F= IRQ15 Bits[7:4] Reserved Note: An Interrupt is activated by enabling this device (offset 30h), setting this register to a non-zero value and setting any combination of bits 0-4 in the corresponding UART IER and the occurrence of the corresponding UART event (for example, Modem Status Change, Receiver Line Error Condition, Transmit Data Request, Receiver Data Available or Receiver Time Out) and setting the OUT2 bit in the MCR.
Reserved	74h	Bit 7:0 - Reserved
Reserved	75h	Bit 7:0 - Reserved
Reserved	F0h	Bit 7:0 - Reserved

Table 15-33. Logical Device 5 (Serial Port 2) (Sheet 1 of 2)

Logical Device Register	Address	Description

Enable
Default = 00h
	30h
	(RW)
		Bits[7:1] Reserved, set to zero.
		Bit[0]
		1 = enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive

I/O Base Address Default = 00h	60h (RW) 61h (Bits 7:3 RW Bits 2:0 RO)	Registers 60h (MSB) and 61h (LSB) set the base address for the device. Note: Decode is on 8 Byte boundaries Comm Decode Ranges 3F8 - 3FF (COM 1) 2F8 - 2FF (COM 2) 220 - 227 228 - 22F 238 - 23F 2E8 - 2EF (COM 4) 338 - 33F 3E8 - 3EF (COM 3)

Table 15-33. Logical Device 5 (Serial Port 2) (Sheet 2 of 2)

Logical Device Register	Address	Description
Primary Interrupt Select Default = 00h	70h (RW)	Bits[3:0] selects which interrupt level is used for the primary Interrupt. 00= no interrupt selected 01= IRQ1 02= IRQ2 03= IRQ3 04= IRQ4 05= IRQ5 06= IRQ6 07= IRQ7 08= IRQ8 09= IRQ9 0A= IRQ10 0B= IRQ11 0C= IRQ12 0D= IRQ13 0E= IRQ14 0F= IRQ15 Bits[7:4] Reserved Note: An Interrupt is activated by enabling this device (offset 30h), setting this register to a non-zero value and setting any combination of bits 0-4 in the corresponding UART IER and the occurrence of the corresponding UART event (for example, Modem Status Change, Receiver Line Error Condition, Transmit Data Request, Receiver Data Available or Receiver Time Out) and setting the OUT2 bit in the MCR.
Reserved	74h	Bit 7:0 - Reserved
Reserved	75h	Bit 7:0 - Reserved
Reserved	F0h	Bit 7:0 - Reserved

Table 15-34. Logical Device 6 (Watch Dog Timer)

Logical Device Register	Address	Description

Enable
Default = 00h
	30h
	(RW)
		Bits[7:1] Reserved, set to zero.
		Bit[0]
		1 = enable the logical device currently selected through the Logical Device # register.
		0 = Logical device currently selected is inactive

I/O Base Address Default = 00h	60h (RW) 61h (Bits 7:5 RW Bits 4:0 RO)	Registers 60h (MSB) and 61h (LSB) set the base address for the device. Note: Decode is on 32 Byte boundaries. This Base Address must be within the 128 bytes of LG1 Base register. Also the last byte accessed by the WDT must not exceed the LG1 Base Address +128 bytes. See Section 15.4.1.2, "Logical Device Configuration Registers [30h — FFh]" for example on how to program.
Primary Interrupt Select Default = 00h	70h (RW)	Bits[3:0] selects which interrupt level is used for the primary Interrupt. 00= no interrupt selected 01= IRQ1 02= IRQ2 03= IRQ3 04= IRQ4 05= IRQ5 06= IRQ6 07= IRQ7 08= IRQ8 09= IRQ9 0A= IRQ10 0B= IRQ11 0C= IRQ12 0D= IRQ13 0E= IRQ14 0F= IRQ15 Bits[7:4] Reserved ***Note:*** An Interrupt is activated by enabling this device (offset 30h), setting this register to a non-zero value and when the first stage has been allowed to reach zero. An Interrupt is not generated if WDT_TOUT_CNF is set to change output after every timeout (See WDT Lock Register).

§ §

16.0 Per Thread WDT (B0:D31:F7)

This is the Watchdog Host Controller device to implement per thread reset capability. The controller manifests in the PCH as a PCI Express* legacy EndPoint. This device is intended to be MSI-X compatible and the design/validation should reflect this compatibility.

16.1 PCI Registers

This section provides details on list of registers that will be supported.

16.2 Register Attribute Legends

The following terminology is used in the definition of the register attribute in this section:

Register Attribute	Meaning
RO	Read only register
RO/V	Read only register / value is variable
RW	Read-Write register
RWS	RW with special features
WO/V	Write only register / value is variable
RO/S	Read only sticky (only reset by loss of power)

16.3 Configuration Space

All of the WDT configuration registers are in the core well. All registers not mentioned are reserved. The configuration space of the PCI registers is accessed through configuration cycle type 0, and it is accessible only as DW (32b) granularity (with byte enables).

16.3.1 PCI Header (B0:D31:F7)

Table 16-1. PCI Header (B0:D31:F7)

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: ID—Identifiers" on page 754	8086h
04h	05h	"Offset 04h: CMD—Command" on page 755	0000h
06h	07h	"Offset 06h: DSTS—Device Status" on page 756	0011h
08h	08h	"Offset 08h: RID—Revision ID" on page 757	See register description
09h	09h	"Offset 09h: CC—Class Code" on page 757	088000h
0Ch	0Ch	"Offset 0Ch: CLS—Cache Line Size" on page 758	00h
0Dh	0Dh	"Offset 0Dh: MLT—Master Latency Timer" on page 758	00h
0Eh	0Eh	"Offset 0Eh: HTYPE—Header Type" on page 758	00h
10h	13h	"Offset 10h: TBAR—WDT Table Base Address" on page 759	00000000h
14h	17h	"Offset 14h: PBAR—WDT PBA Base Address" on page 759	00000000h
18h	1Ch	"Offset 18h: CBAR—WDT CFG Base Address" on page 760	00000000h
2Dh	2Dh	"Offset 2Ch: SSVID—Subsystem Vendor ID" on page 760	XXh
2Eh	2Eh	"Offset 2Eh: SSID Subsystem ID" on page 761	XXh
34h	34h	"Offset 34h: CAP—Capabilities Pointer" on page 761	80h

16.3.1.1 Offset 00h: ID—Identifiers

Table 16-2. Offset 00h: ID—Identifiers

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 23608086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	DID	Device Identification — Indicates the device ID assigned to this PCI device. DID = 2360h		2360h	RO/V
15 :00	VID	Vendor Identification — This 16-bit value is assigned to Intel. Intel VID = 8086h		8086h	RO

16.3.1.2 Offset 04h: CMD—Command

Table 16-3. Offset 04h: CMD—Command

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	Reserved	Reserved		0b	
10	ID	Interrupt Disable — This bit has no effect.		0b	RW
09 :03	Reserved	Reserved		0b	
02	BME	Bus Master Enable — Must be set to '1' for the device to generate PCI Message Signaled Interrupts (MSI). Acts as global interrupt mask. This bit does not affect a device's ability to generate completions as a target.		0b	RW
01	MSE	Memory Space Enable — Enables memory space on the device.		0b	RW
00	IOSE	I/O Space Enable — Enables IO space on the device. Hardwired to 0 as no I/O space implemented.		0b	RO

16.3.1.3 Offset 06h: DSTS—Device Status

Table 16-4. Offset 06h: DSTS—Device Status

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0011h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	SFE	Signaled Fatal Error — This register is set when a device sends an ERR_FATAL message. For Bridges this register is set either because the bridge was the detecting agent of a fatal error and fatal error reporting is enabled, or when forwarding a ERR_FATAL message from secondary bus to primary bus. Note: An ERR_FATAL message is sent when the SFE register is set to a '1' (set event) independent of the present state of the SFE register. This register is only reset by a loss of power. Hardwired to 0 as no error reporting is implemented.		0b	RO
14	SNE	Signaled Non-Fatal Error — This register is set when a device sends an ERR_NONFATAL message. For Bridges this register is set either because the bridge was the detecting agent of a Non-Fatal error and Non-Fatal error reporting is enabled, or when forwarding a ERR_NONFATAL message from secondary bus to primary bus. Note: An ERR_NONFATAL message is sent when the SNE register is set to a '1' (set event) independent of the present state of the SNE register. This register is only reset by a loss of power. Hardwired to 0 as no error reporting is implemented.		0b	RO
13	SPT	Signaled Poisoned TLP — PCI-M Type 0 Devices: This register is set when a device sends a TLP with the EP bit set. For Bridges this register is set either because the bridge was the detecting agent of a poisoned TLP (independent of any enable bits) targeting the primary bus, or when forwarding a poisoned TLP from secondary bus to primary bus. This register is only reset by a loss of power. Hardwired to 0 as no error reporting is implemented.		0b	RO
12	RUR	Received Unsupported Request — This register is set when a device receives a request that it doesn't support. This register is only reset by a loss of power. Hardwired to 0 as no error reporting is implemented.		0b	RO
11	RUC	Received Unsuccessful Completion — This register is set when a device receives a completion with unsuccessful completion status. Note: This register is used for tracking purposes only. No error message is generated in response to setting this bit. This register is only reset by a loss of power.		0b	RO
10	RPT	Received Poisoned TLP — This register is set when a device receives a downstream TLP with the EP bit set. Note: This register is used for tracking purposes only. No error message is generated in response to setting this bit. This register is only reset by a loss of power.		0b	RO/S
09 :05	Reserved	Reserved		00h	

Table 16-4. Offset 06h: DSTS—Device Status

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0011h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
04	CL	Capabilities List — Indicates presence of capabilities list.		1	RO
03	IS	Interrupt Status — Reflects the state of INTx# messages. This bit is set when the interrupt is to be asserted. This bit is a 0 after the interrupt is cleared (independent of the state of CMD.ID).		0b	RO
02 :00	Reserved	Reserved		1b	

16.3.1.4 Offset 08h: RID—Revision ID

Table 16-5. Offset 08h: RID—Revision ID

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** See register description			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID — Indicates stepping of the host controller hardware.		00h	RO

16.3.1.5 Offset 09h: CC—Class Code

Table 16-6. Offset 09h: CC—Class Code

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 09h **Offset End:** 09h	
Size: 24 bit	**Default:** 088000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16	BCC	Base Class Code — Device Base Class Code This field is reserved in this version of the specification.		08h	RO
15 :08	SCC	Sub-Class Code — Device Sub-Class Code This field is reserved in this version of the specification.		80h	RO
07 :00	PI	Programming Interface — Device Programming Interface. This field is reserved in this version of the specification.		00h	RO

16.3.1.6 Offset 0Ch: CLS—Cache Line Size

Table 16-7. Offset 0Ch: CLS—Cache Line Size

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CLS	Cache Line Size — This register has no meaning for this device as it lives on DMI.		00h	RO

16.3.1.7 Offset 0Dh: MLT—Master Latency Timer

Table 16-8. Offset 0Dh: MLT—Master Latency Timer

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MLT	Master Latency Timer — This register has no meaning as the device lives on DMI.		00h	RO

16.3.1.8 Offset 0Eh: HTYPE—Header Type

Table 16-9. Offset 0Eh: HTYPE—Header Type

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	MFD	Multi-function Device — Indicates this controller is not part of a multi-function device.		0b	RO
06 :00	HL	Header Layout — Indicates that the controller uses a target device layout.		00h	RO

16.3.1.9 Offset 10h: TBAR—WDT Table Base Address

Table 16-10. Offset 10h: TBAR—WDT Table Base Address

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :11	MBA	Memory Base Address — Software programs this register with the base address of the device's memory region. This allows for 2KB of memory		0000h	RW
10 :04	Reserved	Reserved		00h	
:03	PF	PF — Indicates that this range is not pre-fetchable		0b	RO
02 :01		Width — This is a 32 bit register		00b	RO
00	MS	Memory Space — Set to 0 for Memory Space BAR's.		0b	RO

16.3.1.10 Offset 14h: PBAR—WDT PBA Base Address

Table 16-11. Offset 14h: PBAR—WDT PBA Base Address

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	MBA	Memory Base Address — Software programs this register with the base address of the device's memory region. This allows for 128 Bytes of memory		0000h	RW
06 :04	Reserved	Reserved		00h	
03	PF	PF — Indicates that this range is not pre-fetchable		0b	RO
02 :01		Width — This is a 32 bit register		00b	RO
00	MS	Memory Space — Set to 0 for Memory Space BAR's.		0b	RO

16.3.1.11 Offset 18h: CBAR—WDT CFG Base Address

Table 16-12. Offset 18h: CBAR—WDT CFG Base Address

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31 :F7	**Offset Start:** 18h **Offset End:** 1Ch	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	MBA	Memory Base Address — Software programs this register with the base address of the device's memory region. This allows for 128 Bytes of memory		0000h	RW
09 :04	Reserved	Reserved		00h	
03	PF	PF — Indicates that this range is not pre-fetchable		0b	RO
02 :01		Width — This is a 32 bit register		00b	RO
00	MS	Memory SpaceSSVID: Set to 0 for Memory Space BAR's.		0b	RO

16.3.1.12 Offset 2Ch: SSVID—Subsystem Vendor ID

This register, in combination with the Subsystem ID register, enables the operating system to distinguish each subsystem from the others. There is no reset for this register.

Table 16-13. Offset 2Ch: SSVID—Subsystem Vendor ID

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31 :F7	**Offset Start:** 2Dh **Offset End:** 2Dh	
Size: 8 bit	**Default:** XXh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SSVID	Subsystem Vendor ID		XXh	RWO

16.3.1.13 Offset 2Eh: SSID Subsystem ID

BIOS sets the value in this register to identify the Subsystem ID. This register, in combination with the Subsystem Vendor ID register, enables the operating system to distinguish each subsystem from other(s). There is no reset for this register.

Table 16-14. Offset 2Eh: SSID Subsystem ID

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 2Eh **Offset End:** 2Eh	
Size: 8 bit	**Default:** XXh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SSID	Subsystem ID		XXh	RWO

16.3.1.14 Offset 34h: CAP—Capabilities Pointer

Table 16-15. Offset 34h: CAP—Capabilities Pointer

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 80h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	Capability Pointer — Indicates that the first capability pointer offset is offset 80h (the Message Signaled Interrupt capability).		80h	RO

16.3.2 PCI MSI-X Capability

Table 16-16. PCI MSI-X Capability

Offset Start	Offset End	Register ID - Description	Default Value
80h	81h	"Offset 80h: MID—MSI-X Capability ID" on page 762	0011h
82h	83h	"Offset 82h: MC—MSI-X Control & Status" on page 762	007Fh
84h	87h	"Offset 84h: MT—MSI-X Table" on page 762	00000000h
88h	8Bh	"Offset 88h: MP—MSI-X Message PBA" on page 763	00000001h

16.3.2.1 Offset 80h: MID—MSI-X Capability ID

Table 16-17. Offset 80h: MID—MSI-X Capability ID

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 80h **Offset End:** 81h	
Size: 8 bit	**Default:** 0011h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — The location of the next capability structure. Hardwired to 00h indicating that this is the last capability header.		000h	RO
07 :00	CID	Capability ID — Capabilities ID indicates MSI-X.		11h	RO

16.3.2.2 Offset 82h: MC—MSI-X Control & Status

Table 16-18. Offset 82h: MC—MSI-X Control & Status

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000007Fh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15		Enable — No interrupts sent unless this bit is set		0b	RW
14		Mask — No effect		0b	RW
13 :11	Reserved	Reserved		0h	
10 :00		Table Size — Size of MSI-X Table. Set to 0x7F for 128 interrupts		7Fh	RO

16.3.2.3 Offset 84h: MT—MSI-X Table

Table 16-19. Offset 84h: MT—MSI-X Table

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :03		Offset — location in BAR of the MSI-X Table		0000h	RO
02 :00	BIR	Indicates BAR 0 at 0x10		0h	RO

16.3.2.4 Offset 88h: MP—MSI-X Message PBA

Table 16-20. Offset 88h: MP—MSI-X Message PBA

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 88h **Offset End:** 8Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :03		Offset — location in BAR of the MSI-X Table		0000h	RO
02 :00	BIR	Indicates BAR 0 at 0x14		1h	RO

16.4 MMIO Space

The MMIO space of the PCI registers is accessed through memory cycle type, and it is accessible only at DW (32b) granularity (with byte enables). It does not support burst accesses to the register sets.

Implementation note: It is expected that design will code the WDT section in a parameterisable manner such that the number of WDTs can be easily varied. The PCH will have a total of 64 WDTs.

The configuration of the WDT device has been formatted to fit into an MSI-X compatible device for possible future compatibility with mainstream OS.

16.4.1 Table BAR Range

There are eight logical registers for each WDT (four for each interrupt) to comply with the MSI specification. The detail for the first eight CSRs are provided in Table 16-21 below. The register layout for the other WDTs in the BAR is identical.

Note: *Some registers in this BAR are aliases to each other. This means that the physical register exists at two different offsets and can be read/written at two different addresses. System software only needs to write to one of the addresses in such cases.*

Other registers are reserved/hardwired to zero – this means they always read as '0x00000000' and write have no effect.

The entire Table BAR contents become read only if WDT_GBLCFG.T_LOCK is set, and only a hardware reset will unlock the contents.

The BAR has been designed to be largely MSI-X compatible but the aliases and reserved registers violate this – necessary to reduce the impact to floorplan.

The descriptions in the expected usage fields hereafter are for situations where interrupt remapping is disabled in the core, which is the expected usage for the PCH.

Table 16-21. Table BAR Range

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: WDT0MAW—WDT 0 Msg Add WARN" on page 765	FEE00000h
04h	07h	"Offset 04h: WDT0MUAW—WDT 0 Msg Upper Add WARN" on page 765	00000000h
08h	0Bh	"Offset 08h: WDT0MDW—WDT 0 Msg Data WARN" on page 766	00004000h
0Ch	0Fh	"Offset 0Ch: WDT0VCW—WDT 0 Vector Ctrl WARN" on page 766	00000000h
10h	13h	"Offset 10h: WDT0MAR—WDT 0 Msg Add RESET" on page 766	See Register Description
14h	17h	"Offset 14h: WDT0MUAR—WDT 0 Msg Upper Add RESET" on page 766	00000000h
18h	1Bh	"Offset 18h: WDT0MDR—WDT 0 Msg Data RESET" on page 767	00004000h
1Ch	1Fh	"Offset 1Ch: WDT0VCR—WDT 0 Vector Ctrl RESET" on page 767	00000000h
		... and so on for a total of 64 WDTs	
7E0h	7E3h	WDT63MAW - WDT 63 Message Address Warn	
7E4h	7E7h	WDT63MUAW - WDT 63 Message Upper Address Warm	
7E8h	7EBh	WDT63MDW - WDT 63 Message Data Warn	
7ECh	7EFh	WDT63VCW - WDT 63 Vector Control Warn	
7F0h	7F3h	WDT63MAR - WDT 63 Message Address Reset	
7F4h	7F7h	WDT63MUAR - WDT 63 Message Upper Address Reset	
7F8h	7FBh	WDT63MDR - WDT 63 Message Data Reset	
7FCh	7FFh	WDT63VCR - WDT 63 Vector Control Reset	

16.4.1.1 Offset 00h: WDT0MAW—WDT 0 Msg Add WARN

Table 16-22. Offset 00h: WDT0MAW—WDT 0 Msg Add WARN

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** FEE00000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20		Address of local APIC area		FEEh	RO
19 :12	APICID	APICID — for local APIC identification		00h	RW
11 :04		Extended APICID — for local APIC identification		00h	RW
03		Should be set to 0		0h	RW
02		As WDT0MDW[11] – must be set to same value 0 => physical mode 1 => logical mode		0h	RW
01 :00	Reserved	Reserved		0h	

16.4.1.2 Offset 04h: WDT0MUAW—WDT 0 Msg Upper Add WARN

This register is hardwired to zero.

16.4.1.3 Offset 08h: WDT0MDW—WDT 0 Msg Data WARN

Table 16-23. Offset 08h: WDT0MDW—WDT 0 Msg Data WARN

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 08h **Offset End:** 0Bh	
Size: 32 bit	**Default:** 00004000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved		0000h	
15		Should be set to 0		0h	RW
14		Should be set to 1		1h	RW
13 :12		Should be set to 0		0h	RW
11		As WDT0MAW[2] – must be set to same value 0 => physical mode 1 => logical mode		0h	RW
10 :08		Delivery Mode for Warning interrupt		0h	RW
07 :00		Vector for interrupt		0h	RW

16.4.1.4 Offset 0Ch: WDT0VCW—WDT 0 Vector Ctrl WARN

Table 16-24. Offset 0Ch: WDT0VCW—WDT 0 Vector Ctrl WARN

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved		0000h	
00	MASK	Masks the Warning Interrupt. Corresponding PBA bit will still get set but the interrupt will not be delivered.		0h	RW

Note: The other WDT (1-63) MSI WARN registers are defined in like manner

16.4.1.5 Offset 10h: WDT0MAR—WDT 0 Msg Add RESET

This register is an alias of WDT0MAW.

16.4.1.6 Offset 14h: WDT0MUAR—WDT 0 Msg Upper Add RESET

This register is hardwired to zero.

16.4.1.7 Offset 18h: WDT0MDR—WDT 0 Msg Data RESET

Table 16-25. Offset 18h: WDT0MDR—WDT 0 Msg Data RESET

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00004000h			**Power Well:**	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved		0000h	
15		Should be set to 0		0h	RW
14		Should be set to 1		1h	RW
13 :12		Should be set to 0		0h	RW
11		As WDT0MAW[2] – must be set to same value 0 => physical mode; 1 => logical mode		0h	RW
10 :08		Delivery Mode for RESET interrupt		0h	RW
07 :00		Vector for interrupt		0h	RW

16.4.1.8 Offset 1Ch: WDT0VCR—WDT 0 Vector Ctrl RESET

Table 16-26. Offset 1Ch: WDT0VCR—WDT 0 Vector Ctrl RESET

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved		0000h	
00	MASK	Masks the RESET Interrupt. Corresponding PBA bit will still get set but the interrupt will not be delivered.		0h	RW

Note: The other WDT (1-63) MSI RESET registers are defined in like manner.

16.4.2 PBA BAR Range

PBA stands for pending bit array. This BAR complies with the MSI specification. This is a minimum size BAR as the array is small.

The pending bits are set to note that RESET/WARN interrupts are pending. The bits will be set when the RESET/WARN event occurs even if the corresponding interrupt is masked. The pending bits will be cleared when the corresponding interrupt is sent or when the WDT for the RESET/WARN pair is reloaded. Setting the RESET bit will clear the corresponding WARN bit. SW cannot write any of the bits.

Table 16-27. PBA BAR Range

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: WDT_PBA0—Pending Bit Array 0" on page 769	00000000h
04h	07h	"Offset 04h: WDT_PBA1—Pending Bit Array 1" on page 770	00000000h
08h	0Bh	"Offset 08h: WDT_PBA2—Pending Bit Array 2" on page 771	00000000h
0Ch	0Fh	"Offset 0Ch: WDT_PBA3—Pending Bit Array 3" on page 772	00000000h

16.4.2.1 Offset 00h: WDT_PBA0—Pending Bit Array 0

Table 16-28. Offset 00h: WDT_PBA0—Pending Bit Array 0

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Pending Bit for RESET 15		0h	RO
30		Pending Bit for WARN 15		0h	RO
29		Pending Bit for RESET 14		0h	RO
28		Pending Bit for WARN 14		0h	RO
27		Pending Bit for RESET 13		0h	RO
26		Pending Bit for WARN 13		0h	RO
25		Pending Bit for RESET 12		0h	RO
24		Pending Bit for WARN 12		0h	RO
23		Pending Bit for RESET 11		0h	RO
22		Pending Bit for WARN 11		0h	RO
21		Pending Bit for RESET 10		0h	RO
20		Pending Bit for WARN 10		0h	RO
19		Pending Bit for RESET 9		0h	RO
18		Pending Bit for WARN 9		0h	RO
17		Pending Bit for RESET 8		0h	RO
16		Pending Bit for WARN 8		0h	RO
15		Pending Bit for RESET 7		0h	RO
14		Pending Bit for WARN 7		0h	RO
13		Pending Bit for RESET 6		0h	RO
12		Pending Bit for WARN 6		0h	RO
11		Pending Bit for RESET 5		0h	RO
10		Pending Bit for WARN 5		0h	RO
09		Pending Bit for RESET 4		0h	RO
08		Pending Bit for WARN 4		0h	RO
07		Pending Bit for RESET 3		0h	RO
06		Pending Bit for WARN 3		0h	RO
05		Pending Bit for RESET 2		0h	RO
04		Pending Bit for WARN 2		0h	RO
03		Pending Bit for RESET 1		0h	RO
02		Pending Bit for WARN 1		0h	RO
01		Pending Bit for RESET 0		0h	RO
00		Pending Bit for WARN 0		0h	RO

16.4.2.2 Offset 04h: WDT_PBA1—Pending Bit Array 1

Table 16-29. Offset 04h: WDT_PBA1—Pending Bit Array 1

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Pending Bit for RESET 31		0h	RO
30		Pending Bit for WARN 31		0h	RO
29		Pending Bit for RESET 30		0h	RO
28		Pending Bit for WARN 30		0h	RO
27		Pending Bit for RESET 29		0h	RO
26		Pending Bit for WARN 29		0h	RO
25		Pending Bit for RESET 28		0h	RO
24		Pending Bit for WARN 28		0h	RO
23		Pending Bit for RESET 27		0h	RO
22		Pending Bit for WARN 27		0h	RO
21		Pending Bit for RESET 26		0h	RO
20		Pending Bit for WARN 26		0h	RO
19		Pending Bit for RESET 25		0h	RO
18		Pending Bit for WARN 25		0h	RO
17		Pending Bit for RESET 24		0h	RO
16		Pending Bit for WARN 24		0h	RO
15		Pending Bit for RESET 23		0h	RO
14		Pending Bit for WARN 23		0h	RO
13		Pending Bit for RESET 22		0h	RO
12		Pending Bit for WARN 22		0h	RO
11		Pending Bit for RESET 21		0h	RO
10		Pending Bit for WARN 21		0h	RO
09		Pending Bit for RESET 20		0h	RO
08		Pending Bit for WARN 20		0h	RO
07		Pending Bit for RESET 19		0h	RO
06		Pending Bit for WARN 19		0h	RO
05		Pending Bit for RESET 18		0h	RO
04		Pending Bit for WARN 18		0h	RO
03		Pending Bit for RESET 17		0h	RO
02		Pending Bit for WARN 17		0h	RO
01		Pending Bit for RESET 16		0h	RO
00		Pending Bit for WARN 16		0h	RO

16.4.2.3 Offset 08h: WDT_PBA2—Pending Bit Array 2

Table 16-30. Offset 08h: WDT_PBA2—Pending Bit Array 2

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 08h **Offset End:** 0Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Pending Bit for RESET 47		0h	RO
30		Pending Bit for WARN 47		0h	RO
29		Pending Bit for RESET 46		0h	RO
28		Pending Bit for WARN 46		0h	RO
27		Pending Bit for RESET 45		0h	RO
26		Pending Bit for WARN 45		0h	RO
25		Pending Bit for RESET 44		0h	RO
24		Pending Bit for WARN 44		0h	RO
23		Pending Bit for RESET 43		0h	RO
22		Pending Bit for WARN 43		0h	RO
21		Pending Bit for RESET 42		0h	RO
20		Pending Bit for WARN 42		0h	RO
19		Pending Bit for RESET 41		0h	RO
18		Pending Bit for WARN 41		0h	RO
17		Pending Bit for RESET 40		0h	RO
16		Pending Bit for WARN 40		0h	RO
15		Pending Bit for RESET 39		0h	RO
14		Pending Bit for WARN 39		0h	RO
13		Pending Bit for RESET 38		0h	RO
12		Pending Bit for WARN 38		0h	RO
11		Pending Bit for RESET 37		0h	RO
10		Pending Bit for WARN 37		0h	RO
09		Pending Bit for RESET 36		0h	RO
08		Pending Bit for WARN 36		0h	RO
07		Pending Bit for RESET 35		0h	RO
06		Pending Bit for WARN 35		0h	RO
05		Pending Bit for RESET 34		0h	RO
04		Pending Bit for WARN 34		0h	RO
03		Pending Bit for RESET 33		0h	RO
02		Pending Bit for WARN 33		0h	RO
01		Pending Bit for RESET 32		0h	RO
00		Pending Bit for WARN 32		0h	RO

16.4.2.4 Offset 0Ch: WDT_PBA3—Pending Bit Array 3

Table 16-31. Offset 0Ch: WDT_PBA3—Pending Bit Array 3

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31:F7		**Offset Start:** 0Ch **Offset End:** 0Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Pending Bit for RESET 63		0h	RO
30		Pending Bit for WARN 63		0h	RO
29		Pending Bit for RESET 62		0h	RO
28		Pending Bit for WARN 62		0h	RO
27		Pending Bit for RESET 61		0h	RO
26		Pending Bit for WARN 61		0h	RO
25		Pending Bit for RESET 60		0h	RO
24		Pending Bit for WARN 60		0h	RO
23		Pending Bit for RESET 59		0h	RO
22		Pending Bit for WARN 59		0h	RO
21		Pending Bit for RESET 58		0h	RO
20		Pending Bit for WARN 58		0h	RO
19		Pending Bit for RESET 57		0h	RO
18		Pending Bit for WARN 57		0h	RO
17		Pending Bit for RESET 56		0h	RO
16		Pending Bit for WARN 56		0h	RO
15		Pending Bit for RESET 55		0h	RO
14		Pending Bit for WARN 55		0h	RO
13		Pending Bit for RESET 54		0h	RO
12		Pending Bit for WARN 54		0h	RO
11		Pending Bit for RESET 53		0h	RO
10		Pending Bit for WARN 53		0h	RO
09		Pending Bit for RESET 52		0h	RO
08		Pending Bit for WARN 52		0h	RO
07		Pending Bit for RESET 51		0h	RO
06		Pending Bit for WARN 51		0h	RO
05		Pending Bit for RESET 50		0h	RO
04		Pending Bit for WARN 50		0h	RO
03		Pending Bit for RESET 49		0h	RO
02		Pending Bit for WARN 49		0h	RO
01		Pending Bit for RESET 48		0h	RO
00		Pending Bit for WARN 48		0h	RO

16.4.3 CFG BAR Range

Table 16-32. CFG BAR Range

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"Offset 00h: WDT0COUNT—WDT Count Register" on page 773	00000000h
04h	07h	"Offset 04h: WDT0CMD—WDT Cmd/Status Register" on page 774	00000000h
.... and so on for a total of 64 WDTs			
1F8h	1FBh	WDT63COUNT - WDT Count Register	
1FCh	1FFh	WDT63CMD - WDT Cmd/Stat Register	
Global Config Section starts at address 0x200			
200h	203h	"Offset 200h: WDT_PSCALE" on page 774	00000000h
204h	207h	"Offset 204h: WDT_GBLCFG" on page 775	00000000h

16.4.3.1 Offset 00h: WDT0COUNT—WDT Count Register

Table 16-33. Offset 00h: WDT0COUNT—WDT Count Register

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	COUNT	Counter Reload Value for bits [31:10] of counter reloading is disabled if this value is zero. Events (setting PBA bit and interrupts) will not trigger if this value is zero. Value can also be locked setting LOCK bit.		000h	RWS
09 :01	Reserved	Reserved – the lower 10 bits of the reload value are always zero		00h	
00	LOCK	Locks bits [31:10] when = 1. Writes of 0 have no effect, requires a reset to unlock the COUNT value.		0h	RWS

16.4.3.2 Offset 04h: WDT0CMD—WDT Cmd/Status Register

Any write of any value to this registers will clear its contents and will also reload the WDT timer (if the WDTnCOUNT.COUNT reload value is non-zero).

Table 16-34. Offset 04h: WDT0CMD—WDT Cmd/Status Register

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31 :F7		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved		00h	
01	RT	RESET interrupt triggered. This gets set when the RESET interrupt message is issued. SW resets by any write to this register.		0h	RWS
00	WT	WARN interrupt triggered. This gets set when the WARN interrupt message is issued. SW resets by any write to this register.		0h	RWS

Note: 63 further pairs of WDT Count/Cmd registers exist at addresses to 0x13C and are all defined in an identical manner.

16.4.3.3 Offset 200h: WDT_PSCALE

The prescaler counter runs as a free running counter. It cannot be disabled. The counter rolls over (and decrements the operating WDT counters) when this value is reached.

Table 16-35. Offset 200h: WDT_PSCALE

Description:					
View: PCI	**BAR:**	**Bus:Device:Function:** B0:D31 :F7		**Offset Start:** 200h **Offset End:** 203h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	LOCK	Locks bits [9:0] when = 1. Writes of 0 have no effect. Requires a reset to unlock the prescaler.		0h	RWS
30 :10	Reserved	Reserved		00h	
09 :00		Prescaler Value applied to all Watchdog timers. Read Only when bit 31 = '1'.		0h	RWS

16.4.3.4 Offset 204h: WDT_GBLCFG

Table 16-36. Offset 204h: WDT_GBLCFG

Description:					
View: PCI	**BAR:**		**Bus:Device:Function:** B0:D31:F7	**Offset Start:** 204h **Offset End:** 207h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved		00h	
01	T_LOCK	Makes the entire Table BAR contents read only when set. A write of 1 sets this bit. A write of 0 has no effect. A reset is required to unlock the Table BAR values if BIOS sets this bit.		0h	RWS
00	GEN	Global Enable – no timers will decrement if this bit is '0'.		0h	RWS

§ §

17.0 Thermal Sensor Registers (B0:D31:F6)

The PCH incorporates two on-die thermal sensors for thermal management.Thermal Sensor 0 (TS0) is the sensor located near the DMI and PCI Express interfaces. Thermal Sensor 1 (TS1) is the sensor located near the SATA* and PCIe* Ports. The PCH supports all of the capabilities of the four-function mode but does so in analog mode. Therefore, the PCH supports catastrophic, hot, AUX and AUX2 trip points, interrupts, and transitions (low-high and high-low).

- Hot On: Calculated value is (itsalertthreshon[7:0] - 50°C)
- Hot Off: Calculated value is (itsalertthreshoff[7:0] - 50°C)
- Catastrophic: Calculated value is (itstripthreshon[7:0] - 50°C)
- AUX: Calculated value is (AUX[7:0] - 50°C)
- AUX2: Calculated value is (AUX2[7:0] - 50°C)

17.1 Thermal Configuration Registers

17.1.1 PCI Configuration Topology

The thermal register set is made available to the OS as a PCI Bus device (B0:D31:F6). The bus number is captured for every type 0 configuration write cycles matching the device.function number. The captured bus # is returned as part of the completer ID field of completions.

17.1.2 Configuration Register Access Restrictions

All controller registers (including the memory mapped registers) must be addressable as byte, word, and D-word quantities. The software must always make register accesses on natural boundaries; D-word accesses must be on D-word boundaries, word accesses on word boundaries, etc.

Note: The thermal memory-mapped register space must not be accessed with the LOCK semantic exclusive-access mechanism. If software attempts exclusive-access mechanisms to the thermal memory-mapped register space, the results are undefined.

All of the registers are in the core well unless otherwise indicated. All registers not mentioned are reserved. Reserved registers will always read 00h and writes will have no effect. Software must properly handle reserved bits. Reserved bits may be designated ‘RsvdP’ or ‘RsvdZ’. Bits marked ‘RsvdP’ must be preserved using read-modify-writes, while ‘RsvdZ’ bits must be written as zeros. This behavior helps to ensure future compatibility.

17.1.3 Temperature Alert

The TEMP_ALERT# pin is controlled by MPC firmware. For the PCH, it is set to GP49.

See the register descriptions for how to enable this feature.

17.1.4 Register Attribute Legend

The following terminology is used in the definition of the register attribute in this specification:

Register Attribute	Meaning
RO	Read-Only register
RW	Read-Write register
RWC	Read-only status, write-to-clear status register
RW1C	Read-only status, write-1-to-clear status register
ROS	Sticky bit- Read-Only register – not affected by reset
RWS	Sticky bit- Read-Write register – not affected by reset
RW1CS	Sticky bit- Read-only status, Write-1-to-clear status register
HWINIT	Hardware initialized; bits are read-only and cannot be reset
RsvdP	Reserved; software must do a read-modify-write to preserve the value of bits
RsvdZ	Reserved; software must use zeros for writes to bits
RSM	Bit(s) are in the 'Resume Well', for example, they maintain their state in any power state from which the Thermal controller may wake the system and are not affected by reset.

17.2 PCI Configuration Registers (Thermal Sensor - B0:D31:F6)

17.2.1 Thermal Sensor PCI Register Address Map

Table 17-1. Bus 0, Device 31, Function 6: Summary of Thermal Reporting PCI Configuration Registers (Sheet 1 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"Offset 00h: ID: Vendor Identification Register" on page 779	8086h
02h	03h	"Offset 02h: DID—Device Identification Register" on page 779	2332h/23B2h[1]
04h	05h	"Offset 04h: ID—Command Register" on page 780	0000h
06h	07h	"Offset 06h: ID—STS—Status Register" on page 781	0010h
08h	08h	"Offset 08h: RID—Revision Identification Register" on page 782	00h
09h	09h	"Offset 09h: PI—Programming Interface Register" on page 782	00h
0Ah	0Ah	"Offset 0Ah: SCC—Sub Class Code Register" on page 782	80h
0Bh	0Bh	"Offset 0Bh: BCC—Base Class Code Register" on page 783	11h
0Ch	0Ch	"Offset 0Ch: Cache Line Size" on page 783	00h
0Dh	0Dh	"Offset 0Dh: Latency Timer" on page 783	00h
0Eh	0Eh	"Offset 0Eh: Header Type" on page 784	00h
10h	13h	"Offset 10h: Thermal Base" on page 784	00000004h
14h	17h	"Offset 14h: TBARH—Thermal Base High DWord" on page 785	00000000h
2Ch	2Dh	"Offset 2Ch: Subsystem Vendor ID" on page 785	0000h

Table 17-1. Bus 0, Device 31, Function 6: Summary of Thermal Reporting PCI Configuration Registers (Sheet 2 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
2Eh	2Fh	"Offset 2Eh: Subsystem ID" on page 786	0000h
34h	34h	"Offset 34h: Capabilities Pointer" on page 786	50h
3Ch	3Ch	"Offset 3Ch: Interrupt Line" on page 787	00h
3Dh	3Dh	"Offset 3Dh: Interrupt Pin" on page 787	See Description
40h	43h	"Offset 40h: BIOS Assigned Thermal Base Address" on page 788	00000004h
44h	47h	"Offset 44h: BIOS Assigned Thermal Base High DWord" on page 788	00000000h
50h	51h	"Offset 50h: PCI Power Management Capability ID" on page 789	8001h
52h	53h	"Offset 52h: Power Management Capabilities" on page 789	0023h
54h	57h	"Offset 54h: Power Management Control And Status" on page 790	0008h
80h	81h	"Offset 80h: MID—Message Signaled Interrupt Identifiers" on page 791	0005h
82h	83h	"Offset 82h: MC—Message Signaled Interrupt Message Control" on page 791	0000h
84h	87h	"Offset 84h: MA—Message Signaled Interrupt Message Address" on page 792	00000000h
88h	89h	"Offset 88h: MD—Message Signaled Interrupt Message Data" on page 792	0000h
94h	94h	"Offset 94h: CCTS0RD—Thermal Sensor 0 Remote Diode" on page 792	00h
96h	97h	"Offset 96h: CCTS0SSI—Thermal Sensor 0 Sense Stage Inputs" on page 793	0066h
98h	9Bh	"Offset 98h: CCTS0DSAC0—Thermal Sensor 0 Delta Sigma ADC Control 0" on page 794	02010083h
9Ch	9Dh	"Offset 9Ch: CCTS0DSAC1—Thermal Sensor 0 Delta Sigma ADC Control 1" on page 794	0001h
A0h	A3h	"Offset A0h: CCTS0C—Thermal Sensor 0 Calibration" on page 795	00030016h
A4h	A7h	"Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0" on page 795	80000000h
A8h	ABh	"Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1" on page 797	See Description
ACh	ADh	"Offset ACh: CCTS0BP—Thermal Sensor 0 Bandgap Pins" on page 798	001Bh
B4h	B4h	"Offset B4h: CCTS1RD—Thermal Sensor 1 Remote Diode" on page 798	0000h
B6h	B7h	"Offset B6h: CCTS1SSI—Thermal Sensor 1 Sense Stage Inputs" on page 799	0066h
B8h	BBh	"Offset B8h: CCTS1DSAC0—Thermal Sensor 1 Delta Sigma ADC Control 0" on page 799	02010083h
BCh	BDh	"Offset BCh: CCTS1DSAC1—Thermal Sensor 1 Delta Sigma ADC Control 1" on page 800	0001h
C0h	C3h	"Offset C0h: CCTS1C—Thermal Sensor 1 Calibration" on page 800	00030016h
C4h	C7h	"Offset C4h: CCTS1TL0—Thermal Sensor 1 Top Logic 0" on page 801	80999BAAh
C8h	CBh	"Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1" on page 802	See Description
CCh	CDh	"Offset CCh: CCTS1BP—Thermal Sensor 1 Bandgap Pins" on page 803	001Bh
F8h	FBh	"Offset F8h: MANID—Manufacturing/Process " on page 803	00000F00h

1. The values are for the DH89xxCC/DH89xxCL, respectively.

17.2.1.1 Offset 00h: VID—Vendor Identification

Table 17-2. Offset 00h: ID: Vendor Identification Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor Identification: This 16-bit value is assigned to Intel.		8086h	RO

17.2.1.2 Offset 02h: DID—Device Identification

Table 17-3. Offset 02h: DID—Device Identification Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 2332h/23B2h[1]			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device Identification: This is a 16-bit value assigned by the PCI SIG.		2332h/23B2h[1]	RO

1. The values are for the DH89xxCC/DH89xxCL, respectively.

17.2.1.3 Offset 04h: CMD—Command

Table 17-4. Offset 04h: ID—Command Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		00h	
10	ID	Interrupt Disable: Enables the device to assert an INTx#. When set, the Thermal logic's INTx# signal will be de-asserted. When cleared AND MSI is not enabled, the INTx# signal may be asserted. NOTE: this bit has no affect on MSI generation.		0b	RW
09	FBE	Fast Back to Back Enable: Not Implemented. Hardwired to '0'.		0b	RO
08	SEN	SERR# Enable: Not Implemented. Hardwired to '0'.		0b	RO
07	WCC	Wait Cycle Control: Not Implemented. Hardwired to '0'.		0b	RO
06	PER	Parity Error Response Enable: Not Implemented. Hardwired to '0'.		0b	RO
05	VPS	VGA Palette Snoop: Not Implemented. Hardwired to '0'		0b	RO
04	MWI	Memory Write and Invalidate Enable: Not Implemented. Hardwired to '0'.		0b	RO
03	SCE	Special Cycle Enable: Not Implemented. Hardwired to '0'.		0h	RO
02	BME	Bus Master Enable: When set, enables.		0b	RW
01	MSE	Memory Space Enable: When set, enables memory space accesses to the Thermal registers.		0b	RW
00	IOS	I/O Space Enable: The Thermal logic does not implement IO Space. This bit is Hardwired to '0'.		0b	RO

17.2.1.4 Offset 06h: STS—Status

Table 17-5. Offset 06h: ID—STS—Status Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	Detected Parity Error: 0 = Parity error not detected. 1 = Indicates that the PCH detected a parity error on the internal backbone. This bit gets set even if the CMD.PER bit "Offset 04h: ID—Command Register" bit 6) is not set. Software clears this bit by writing a '1' to this bit location.		0b	RW1C
14	SERRS	SERR# Status: Not Implemented. Hardwired to '0'.		0b	RO
13	RMA	Received Master Abort: Not Implemented. Hardwired to '0'.		0b	RO
12	RTA	Received Target Abort: Not Implemented. Hardwired to '0'.		0b	RO
11	STA	Signaled Target Abort: Not Implemented. Hardwired to '0'.		0b	RO
10 :09	Reserved	Reserved		00b	
08	MDPE	Master Data Parity Error: Not Implemented. Hardwired to '0'.		0b	RO
07	FBC	Fast Back to Back Capable: Does not apply. Hardwired to '0'.		0b	RO
06	Reserved	Reserved		0b	
05	C66	66 MHz Capable: Does not apply. Hardwired to '0'.		0b	RO
04	CLIST	Capabilities List Exists: Indicates the controller contains a capabilities pointer list. The first item is pointed to by looking at configuration offset 34h.		1b	RO
03	IS	Interrupt Status: Reflects the state of the INTx# signal at the input of the enable/disable circuit. This bit is a 1 when the INTx# is asserted. This bit is a 0 after the interrupt is cleared (independent of the state of the Interrupt Disable bit in the command register). This bit is not set if MSI is enabled.		0b	RO
2 :0	Reserved	Reserved		000b	

17.2.1.5 Offset 08h: RID—Revision Identification

Table 17-6. Offset 08h: RID—Revision Identification Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	RID	Revision ID: Indicates the device specific revision identifier. NOTE: The value reported in this register depends on the value of the Revision ID in Device 31, Function 0, Offset 08h. See the LPC section for details.		00h	RO

17.2.1.6 Offset 09h: PI—Programming Interface

Table 17-7. Offset 09h: PI—Programming Interface Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 09h **Offset End:** 09h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PI	Programming Interface — PCH Thermal logic has no standard programming interface.		00h	RO

17.2.1.7 Offset 0Ah: SCC—Sub Class Code

Table 17-8. Offset 0Ah: SCC—Sub Class Code Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 0Ah **Offset End:** 0Ah	
Size: 8 bit	**Default:** 80h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	SCC	Sub Class Code — 8-bit value assigned to PCH Thermal logic.		80h	RO

17.2.1.8 Offset 0Bh: BCC—Base Class Code

Table 17-9. Offset 0Bh: BCC—Base Class Code Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:** 11h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	BCC	Base Class Code — 8-bit value assigned to PCH Thermal logic		11h	RO

17.2.1.9 Offset 0Ch: CLS—Cache Line Size

Table 17-10. Offset 0Ch: Cache Line Size

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CLS	Cache Line Size — Does not apply to PCI Bus Target-only devices.		00h	RO

17.2.1.10 Offset 0Dh: LT—Latency Timer

Table 17-11. Offset 0Dh: Latency Timer

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	LT	Latency Timer — Does not apply to PCI Bus Target-only devices.		00h	RO

17.2.1.11 Offset 0Eh: HTYPE—Header Type

Table 17-12. Offset 0Eh: Header Type

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:07	MFD	Multi-Function Device — This bit is 0 because a multi-function device only needs to be marked as such in Function 0, and the Thermal registers are not in Function 0.		0b	RO
06 :00	HTYPE	Header Type — Implements Type 0 Configuration header.		00h	RO

17.2.1.12 Offset 10h: TBAR—Thermal Base

This BAR creates 4K bytes of memory space to signify the base address of Thermal memory mapped configuration registers. This memory space is active when the Command (CMD) register Memory Space Enable (MSE) bit is set and either TBAR[31:12] or TBARH are programmed to a non-zero address. This BAR is owned by the operating system, and allows the OS to locate the Thermal registers in system memory space.

Table 17-13. Offset 10h: Thermal Base

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	TBA	Thermal Base Address — This field provides the base address for the Thermal logic memory mapped configuration registers. 4 KB bytes are requested by hardwiring bits 11:4 to 0s.		00000h	RW
11 :04	Reserved	Reserved		00h	
03	PREF	Prefetchable — Indicates that this BAR is NOT pre-fetchable.		0b	RO
02 :01	ADDRNG	Address Range — Indicates that this BAR can be located anywhere in 64 bit address space.		10b	RO
00	SPTYP	Space Type — Indicates that this BAR is located in memory space.		0b	RO

17.2.1.13 Offset 14h: TBARH—Thermal Base High DWord

This BAR extension holds the high 32 bits of the 64 bit TBAR. In conjunction with TBAR, it creates 4 KB of memory space to signify the base address of thermal memory mapped configuration registers.

Table 17-14. Offset 14h: TBARH—Thermal Base High DWord

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TBAH	Thermal Base Address High — TBAR bits 61:32.		00000000h	RW

17.2.1.14 Offset 2Ch: SVID—Subsystem Vendor ID

This register should be implemented for any function that could be instantiated more than once in a given system,. The SVID register, in combination with the Subsystem ID register, enables the operating environment to distinguish one subsystem from the other(s).

Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset.

Table 17-15. Offset 2Ch: Subsystem Vendor ID

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	SVID — These RWO bits have no PCH functionality.		0000h	RWO

17.2.1.15 Offset 2Eh: SID—Subsystem ID

This register should be implemented for any function that could be instantiated more than once in a given system. The SID register, in combination with the Subsystem Vendor ID register make it possible for the operating environment to distinguish one subsystem from the other(s).

Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SVID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset.

Table 17-16. Offset 2Eh: Subsystem ID

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	SID — These RWO bits have no PCH functionality.		0000h	RWO

17.2.1.16 Offset 34h: CP—Capabilities Pointer

Table 17-17. Offset 34h: Capabilities Pointer

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 50h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	Capability Pointer — Indicates that the first capability pointer offset is offset 50h (Power Management Capability).		50h	RO

17.2.1.17 Offset 3Ch: INTLN—Interrupt Line

Table 17-18. Offset 3Ch: Interrupt Line

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	INTLN	Interrupt Line — PCH hardware does not use this field directly. It is used to communicate to software the interrupt line that the interrupt pin is connected to.		00h	RW

17.2.1.18 Offset 3Dh: INTPN—Interrupt Pin

Table 17-19. Offset 3Dh: Interrupt Pin

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** See Description			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :04	Reserved	Reserved		0h	
03 :00	INTPN	Interrupt Pin: This reflects the value of the Device 31 interrupt pin bits 27:24 (TTIP) in chipset configuration space.		Desc	RO

17.2.1.19 Offset 40h: TBARB—BIOS Assigned Thermal Base Address

This BAR creates 4K bytes of memory space to signify the base address of Thermal memory mapped configuration registers. This memory space is active when TBARB.SPTYPEN is asserted. This BAR is owned by the BIOS, and allows the BIOS to locate the Thermal registers in system memory space. If both TBAR and TBARB are programmed, then the OS and BIOS each have their own independent "view" of the Thermal registers, and must use the TSIU register to denote Thermal registers ownership/availability.

Note: Transactions to TBARB are not affected by BME or MSE settings, only SPTYPEN. In other words, memory reads and writes to TBARB, when SPTYPEN (bit[0]) is '1', are claimed whether MSE is a '1' or '0' or whether the device is in D3 or D0.

Table 17-20. Offset 40h: BIOS Assigned Thermal Base Address

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 40h **Offset End:** 43h	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	TBA	Thermal Base Address — This field provides the base address for the Thermal logic memory mapped configuration registers. 4K B bytes are requested by hardwiring bits 11:4 to 0s.		00000h	RW
11 :04	Reserved	Reserved		00h	
03	PREF	Prefetchable — Indicates that this BAR is NOT pre-fetchable.		0b	RO
02 :01	ADDRNG	Address Range — Indicates that this BAR can be located anywhere in 64 bit address space.		10b	RO
00	SPTYPEN	Space Type Enable: 0 = Disable. 1 = Enable. When set to 1b by software, enables the decode of this memory BAR.		0b	RW

17.2.1.20 Offset 44h: TBARBH—BIOS Assigned Thermal Base High DWord

Table 17-21. Offset 44h: BIOS Assigned Thermal Base High DWord

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TBAH	Thermal Base Address High: TBAR bits 61:32.		00000000h	RW

17.2.1.21 Offset 50h: PID—PCI Power Management Capability ID

Table 17-22. Offset 50h: PCI Power Management Capability ID

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 50h **Offset End:** 51h	
Size: 16 bit	**Default:** 8001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Capability — Indicates that the next capability is MSI.		80h	RO
07 :00	CAP	Capability ID — Indicates that this pointer is a PCI power management capability		01h	RO

17.2.1.22 Offset 52h: PC—Power Management Capabilities

Table 17-23. Offset 52h: Power Management Capabilities

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** 0023h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11		PME_Support — Indicates PME# is not supported		00h	RO
10		D2_Support — The D2 state is not supported.		0b	RO
09		D1_Support — The D1 state is not supported.		0b	RO
08 :06		Aux_Current — PME# from D3COLD state is not supported, therefore this field is 000b.		000b	RO
05	DSI	Device Specific Initialization — Indicates that device-specific initialization is required.		1b	RO
04	Reserved	Reserved		0b	
03	PMEC	PME Clock — Does not apply. Hardwired to 0.		0b	RO
02 :00	VS	Version — Indicates support for Revision 1.2 of the *PCI Power Management Specification.*		011b	RO

17.2.1.23 Offset 54h: PCS—Power Management Control And Status

Table 17-24. Offset 54h: Power Management Control And Status

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 0008h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24		Data — Does not apply. Hardwired to 0s.		00h	RO
23	BPCCE	Bus Power/Clock Control Enable — Hardwired to 0.		0b	RO
22	B23	B2/B3 Support —Does not apply. Hardwired to 0.		0b	RO
21 :16	Reserved	Reserved		00h	
15	PMES	PME Status — This bit is always 0, since this PCI Function does not generate PME#		0b	RO
14 :09	Reserved	Reserved		00h	
08	PMEE	PME Enable — This bit is always zero, since this PCI Function does not generate PME#		0b	RO
07 :04	Reserved	Reserved		0h	
03		No Soft Reset —When set (‘1’), this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.		1b	RO
02	Reserved	Reserved		0b	
01 :00	PS	Power State — This field is used both to determine the current power state of the Thermal controller and to set a new power state. The values are: 00 = D0 state 11 = $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally; however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ states, the Thermal controller’s configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. When software changes this value from the $D3_{HOT}$ state to the D0 state, no internal warm (soft) reset is generated.		00b	RW

17.2.1.24 Offset 80h: MID—Message Signaled Interrupt Identifiers

Table 17-25. Offset 80h: MID—Message Signaled Interrupt Identifiers

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 80h **Offset End:** 81h	
Size: 16 bit	**Default:** 0005h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	NEXT	Next Pointer — Indicates this is the latest pointer		00h	RO
07 :00	CID	Capability ID — Capabilities ID indicates MSI.		05h	RO

17.2.1.25 Offset 82h: MC—Message Signaled Interrupt Message Control

Table 17-26. Offset 82h: MC—Message Signaled Interrupt Message Control

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :08	Reserved	Reserved		00h	
07	C64	64 Bit Address Capable — Capable of generating a 32-bit message only.		0b	RO
06 :04	MME	Multiple Message Enable — These bits are RW for software compatibility, but only one message is ever sent by the root port.		0h	RW
03 :01	MMC	Multiple Message Capable — Only one message is required.		0h	RO
00	MSIE	MSI Enable — If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. CMD.BME must be set for an MSI to be generated. If CMD.BME is cleared, and this bit is set, no interrupts (not even pin based) are generated.		0b	RO

17.2.1.26 Offset 84h: MA—Message Signaled Interrupt Message Address

Table 17-27. Offset 84h: MA—Message Signaled Interrupt Message Address

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 84h **Offset End:** 87h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	Address — Lower 32 bits of the system specified message address, always DW aligned.		0000000h	RW
01 :00	Reserved	Reserved		00b	

17.2.1.27 Offset 88h: MD—Message Signaled Interrupt Message Data

Table 17-28. Offset 88h: MD—Message Signaled Interrupt Message Data

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 88h **Offset End:** 89h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	Data — This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD[15:0]) during the data phase of the MSI memory write transaction.		0000h	RW

17.2.1.28 Offset 94h: CCTS0RD—Thermal Sensor 0 Remote Diode

Table 17-29. Offset 94h: CCTS0RD—Thermal Sensor 0 Remote Diode

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 94h **Offset End:** 94h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved		00h	
00		Enable remote diodes operation. NOTE: There will be a 2.5ms delay from transition of isnsrmten to delivery of valid data on Offset A8h: CCTS0TL1 – Thermal Sensor 0 Top Logic1 otstemperature[8:0].		0b	RW

17.2.1.29 Offset 96h: CCTS0SSI—Thermal Sensor 0 Sense Stage Inputs

Table 17-30. Offset 96h: CCTS0SSI—Thermal Sensor 0 Sense Stage Inputs

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** 96h **Offset End:** 97h	
Size: 16 bit	**Default:** 0066h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved		0b	
14		isnsinternalvrefen — Enable using an internal reference voltage generated by a resistor divider, instead of using external reference voltage		0b	RW
13		isnswastesel — Select the waste current target: 0 = Current from non-used current mirrors flows to the waste diode 1 = Current from non-used current mirrors flows disable		0b	RW
12 :08		isnschopsel[4:0] — Sense stage current ratio configuration for DEM and chopping (See the Sense stage current ratio configuration)		00h	RW
07		isnsextcuren — Enable driving external bias current		0b	RW
06		isnsbiasampen — Sense stage current bias amplifier enable		1b	RW
05 :03		isnscurrentsel[2:0] — Current value select for 4 current sources within the sense stage (See the Sense stage bias current select)		100b	RW
02		isnsextresen — Enable external resistor use for accurate reference current generation		1b	RW
01		isnsen (See the Sense stage enable signal)		1b	RW
00	Reserved	Reserved		0b	

17.2.1.30 Offset 98h: CCTS0DSAC0—Thermal Sensor 0 Delta Sigma ADC Control 0

Table 17-31. Offset 98h: CCTS0DSAC0—Thermal Sensor 0 Delta Sigma ADC Control 0

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32bit	**Default:** 02010083h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		idstiming[15:0] — Delta Sigma phase timing control (See the Delta Sigma Control and Timing Configuration)		0201h	RW
15 :00		idscontrol[15:0] — Delta Sigma features control (See the Delta Sigma Control and Timing Configuration)		0083	RW

17.2.1.31 Offset 9Ch: CCTS0DSAC 1—Thermal Sensor 0 Delta Sigma ADC Control 1

Table 17-32. Offset 9Ch: CCTS0DSAC1—Thermal Sensor 0 Delta Sigma ADC Control 1

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** 9Ch **Offset End:** 9Dh	
Size: 16 bit	**Default:** 0001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :02	Reserved	Reserved		0000h	
01		idshighgain: 0 – default bias current for SD ADC 1 = reduce by 2 bias current for SD ADC.; Increase the amplifier gain, reduce current consumption and bandwidth		0b	RW
00		idsen — DS ADC enable		1b	RW

17.2.1.32 Offset A0h: CCTS0C—Thermal Sensor 0 Calibration

Table 17-33. Offset A0h: CCTS0C—Thermal Sensor 0 Calibration

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00030016h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved		0000h	
17 :16		icalconfigsel[1:0] — Controls for calibration: 00: linearization only 01: do both calibration and linearization calculations 10: do not calibrate and do not linearization 11: calibration calculation only		11b	RW
15 :08		isparectrl[7:0] — Spare bits (See the Spare pin functionality)		00h	RW
07 :00		icalcoarsetune[7:0] — Calibration bits for coarse calibration range selection		16h	RW

17.2.1.33 Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0

Table 17-34. Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 80000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Itsen — Enable thermal sensor Hard-IP when set to 1		1b	RW
30	Reserved	Reserved		0b	
29 :28		itsclksel[1:0]: 00 = 100 Mhz, 01 = **Reserved** 10 = **Reserved** 11 = 400 Mhz		00b	RW
27	Reserved	Reserved		0b	
26 :24	Reserved	Reserved		000b	

Table 17-34. Offset A4h: CCTS0TL0—Thermal Sensor 0 Top Logic 0

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 80000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16		itsalertthreshoff[7:0] — (Hot value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned off. • Default value has been set for 103°C • Calculated value is (itsalertthreshoff [7:0] - 50°C) ***Note:*** For the PCH these bits are not used and have been replaced with "Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd" on page 812 bits 7:0		99h	RW
15 :08		itsalertthreshon[7:0] — (Hot value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned on (See the Parallel Temperature Diagram). • Default value has been set for 105°C • Calculated value is (itsalertthreshon[7:0] - 50°C) ***Note:*** Thermal protection logic associated with this field will always default to A2h = 112°C whenever this field is programmed to a value greater than A2h due to accidental or malicious programming. • For the PCH these bits are not used and have been replaced with "Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point" on page 810 bits 15:8		98h	RW
07 :00		itstripthreshon[7:0] — (Catastrophic value) When temperature reaches the value indicated by this bus, trip signal (otstemperaturetrip) is turned on (See the Parallel Temperature Diagram). • Default value has been set for 112°C • Calculated value is (itstripthreshon[7:0] - 50°C) ***Note:*** Thermal protection logic associated with this field will always default to A2h = 112°C whenever this field is programmed to a value greater than A2h, due to accidental or malicious programming. • For the PCH these bits are not used and have been replaced with "Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point" on page 810 bits 7:0		A2h	RW

17.2.1.34 Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1

Table 17-35. Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:31:6		**Offset Start:** A8h **Offset End:** ABh	
Size: 32 bit	**Default:** See Description			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30		itsfilterrate[1:0] — Choose the CIC filter decimation factor: 00 = Decimate by 64 01 = Decimate by 128 10 = Decimate by 128 11 = Decimate by 256 NOTE: The FIR has a built-in decimation factor of 4, so if the FIR is not bypassed, the total decimation factor is 4 times the value given by itsfilterrate[1:0]		00b	RW
29 :28		itsfiltersel[1:0] — Select the digital filter mode: 00 = Use CIC followed by FIR 01 = CIC only (bypass the FIR) 10 – Reserved 11 = Reserved		00	RW
27	Reserved	Reserved		0b	
26 :24		itsfilterrangesel[2:0] — Range select for filter output bits according to the gain of the TS analog circuit (See the Range select for filter output)		000b	RW
23 :12		itsgammacoeff[11:0] — Linearization coefficient in Ratiometric mode only		0AFh	RW
11 :10		itsoutputbitsel[1:0] — Resolution selection of temperature outputs (otstemperature[8:0]) from calculated temperature: 00 – default (use the default only) 01 = 1 bit shift left 10 = 1 bit shift right 11 = 2 bit shift left		00b	RW
09	Reserved	Reserved		0b	
08 :00		otstemperature[8:0] — Temperature reading where 0 = -50°C, resolution of 0.5°C (See the Parallel Temperature Diagram). ***Note:*** Default value is X it depends on temperature at power on • This temperature at resolution of 1°C can also be read at MMIO location "Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read" on page 810 bits 7:0		XXh	RO

17.2.1.35 Offset ACh: CCTS0BP—Thermal Sensor 0 Bandgap Pins

Table 17-36. Offset ACh: CCTS0BP—Thermal Sensor 0 Bandgap Pins

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** ACh **Offset End:** ADh	
Size: 16 bit	**Default:** 001Bh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :05	Reserved	Reserved		000h	
04		Ibgen — Enable signal for the bandgap circuit block. A logic '1' enabled the BG circuit and a logic '0' disables (power-down) the circuit block		1b	RW
03		Ibgchopen — Enable signal for the bandgap circuit block chopping mode. A logic '1' enables the chopping mode and a logic '0' disables the chopping mode. A logic '0' disables all of the internal clock generation circuitry.		1b	RW
02		Ibgfilterbypassen — Select signal for the bandgap circuit block chop clock: 0 – Working with LPF 1 = Bypass LPF		0b	RW
01 :00		ibgclksource[1:0]: 00 = DS bit stream 01 - reserve 10 = reserve 11 - internal bandgap clock		11b	RW

17.2.1.36 Offset B4h: CCTS1RD—Thermal Sensor 1 Remote Diode

Table 17-37. Offset B4h: CCTS1RD—Thermal Sensor 1 Remote Diode

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** B4h **Offset End:** B4h	
Size: 8 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved		0000000b	
00		isnsrmten — Enable remote diodes operation. NOTE: There will be a 2.5msec delay from transition of isnsrmten to delivery of valid data on Offset A8h: CCTS0TL1 – Thermal Sensor 0 Top Logic1 otstemperature[8:0].		0b	RW

17.2.1.37 Offset B6h: CCTS1SSI—Thermal Sensor 1 Sense Stage Inputs

Table 17-38. Offset B6h: CCTS1SSI—Thermal Sensor 1 Sense Stage Inputs

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** B6h **Offset End:** B7h	
Size: 16 bit	**Default:** 0066h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved		0b	
14		isnsinternalvrefen — Enable using an internal reference voltage generated by a resistor divider instead of using external reference voltage		0b	RW
13		isnswastesel — Select the waste current target: 0 = Current from non-used current mirrors flows to the waste diode 1 = Current from non-used current mirrors flows disable		0b	RW
12 :08		isnschopsel[4:0] — Sense stage current ratio configuration for DEM and chopping (See the Sense stage current ratio configuration)		00000b	RW
07		isnsextcuren — Enable driving external bias current		0b	RW
06		isnsbiasampen — Sense stage current bias amplifier enable		1b	RW
05 :03		isnscurrentsel[2:0] — Current value select for 4 current sources within the sense stage (See the Sense stage bias current select)		100b	RW
02		isnsextresen — Enable external resistor use for accurate reference current generation		1b	RW
01		isnsen — Sense stage enable signal		1b	RW
00	Reserved	Reserved		0b	

17.2.1.38 Offset B8h: CCTS1DSAC0—Thermal Sensor 1 Delta Sigma ADC Control 0

Table 17-39. Offset B8h: CCTS1DSAC0—Thermal Sensor 1 Delta Sigma ADC Control 0

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 02010083h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		idstiming[15:0] — Delta Sigma phase timing control (See the Delta Sigma Control and Timing Configuration)		0201h	RW
15 :00		idscontrol[15:0] — Delta Sigma features control (See the Delta Sigma Control and Timing Configuration)		0083h	RW

17.2.1.39 Offset BCh: CCTS1DSAC1—Thermal Sensor 1 Delta Sigma ADC Control 1

Table 17-40. Offset BCh: CCTS1DSAC1—Thermal Sensor 1 Delta Sigma ADC Control 1

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** BCh **Offset End:** BDh	
Size: 16 bit	**Default:** 0001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :02	Reserved	Reserved		0000h	
01		idshighgain: 0 = default bias current for SD ADC 1 = reduce by 2 bias current fro SD ADC. Increase the amplifier gain, reduce current consumption and bandwidth		0b	RW
00		idsen — DS ADC enable		1b	RW

17.2.1.40 Offset C0h: CCTS1C—Thermal Sensor 1 Calibration

Table 17-41. Offset C0h: CCTS1C—Thermal Sensor 1 Calibration

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00030016h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved		0000h	
17 :16		icalconfigsel[1:0] — Controls for calibration: 00: linearization only 01: do both calibration and linearization calculations 10: do not calibrate and do not linearization 11: calibration calculation only		11b	RW
15 :08		isparectrl[7:0] — Spare bits (See the Spare pin functionality)		00h	RW
07 :00		icalcoarsetune[7:0] — Calibration bits for coarse calibration range selection		16h	RW

17.2.1.41 Offset C4h: CCTS1TL0—Thermal Sensor 1 Top Logic 0

Table 17-42. Offset C4h: CCTS1TL0—Thermal Sensor 1 Top Logic 0

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 80999BAAh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31		Itsen — Enable thermal sensor Hard-IP when set to 1		1b	RW
30	Reserved	Reserved		0b	
29 :28		itsclksel[1:0]: 00 = 100 Mhz 01 = **Reserved** 10 = **Reserved** 11 = 400 Mhz		00b	RW
27	Reserved	Reserved		0b	RW
26 :24	Reserved	Reserved		000b	
23 :16		itsalertthreshoff[7:0] — (Hot value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned off. • Default value has been set for 103°C • Calculated value is (itsalertthreshoff [7:0] - 50°C) ***Note:*** For the PCH these bits are not used and have been replaced with "Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd" on page 829 bits 7:0		99h	RW
15 :08		itsalertthreshon[7:0] — (Hot value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned on. • Default value has been set for 105°C • Calculated value is (itsalertthreshon[7:0] - 50°C) ***Note:*** Thermal protection logic associated with this field will always default to A2h = 112°C whenever this field is programmed to a value greater than A2h due to accidental or malicious programming. • For the PCH these bits are not used and have been replaced with "Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point" on page 826 bits 15:8		9Bh	RW
07 :00		itstripthreshon[7:0] — (Catastrophic value) When temperature reaches the value indicated by this bus, trip signal (otstemperaturetrip) is turned on. • Default value has been set for 112°C • Calculated value is (itstripthreshon[7:0] - 50°C) ***Note:*** Thermal protection logic associated with this field will always default to A2h = 112°C whenever this field is programmed to a value greater than A2h, due to accidental or malicious programming. • For the PCH these bits are not used and have been replaced with "Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point" on page 826 bits 7:0		A2h	RW

17.2.1.42 Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1

Table 17-43. Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** C8h **Offset End:** CBh	
Size: 32 bit	**Default:** See Description			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30		itsfilterrate[1:0] — Choose the CIC filter decimation factor: 00 = Decimate by 64 01 = Decimate by 128 10 = Decimate by 128 11 = Decimate by 256 NOTE: The FIR has a built-in decimation factor of 4, so if the FIR is not bypassed, the total decimation factor is 4 times the value given by itsfilterrate[1:0]		00b	RW
29 :28		itsfiltersel[1:0] — Select the digital filter mode: 00 = Use CIC followed by FIR 01 = CIC only (bypass the FIR) 10 = Reserved 11 = Reserved		00b	RW
27	Reserved	Reserved		0b	
26 :24		itsfilterrangesel[2:0] — Range select for filter output bits according to the gain of the TS analog circuit		000b	RW
23 :12		itsgammacoeff[11:0] — Linearization coefficient in Ratiometric mode only		0AFh	RW
11 :10		itsoutputbitsel[1:0] — Resolution selection of temperature outputs (otstemperature[8:0]) from calculated temperature: 00 = default (use the default only) 01 = 1 bit shift left 10 = 1 bit shift right 11 = 2 bit shift left		00b	RW
09	Reserved	Reserved		0b	
08 :00		otstemperature[8:0] — Temperature reading, where 0 = -50 ^{o}C, resolution of 0.5 ^{o}C. NOTE: Default value is X it depends on temperature at power on		X	RO

17.2.1.43 Offset CCh: CCTS1BP—Thermal Sensor 1 Bandgap Pins

Table 17-44. Offset CCh: CCTS1BP—Thermal Sensor 1 Bandgap Pins

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** CCh **Offset End:** CDh	
Size: 16 bit	**Default:** 001Bh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :05	Reserved	Reserved		000h	
04		Ibgen — Enable signal for the bandgap circuit block. A logic 1 enabled the BG circuit and a logic 0 disables (power-down) the circuit block		1b	RW
03		Ibgchopen — Enable signal for the bandgap circuit block chopping mode. A logic `1' enables the chopping mode and a logic `0' disables the chopping mode. A logic `0' disables all of the internal clock generation circuitry.		1b	RW
02		Ibgfilterbypassen — Select signal for the bandgap circuit block chop clock.: 0 – Working with LPF 1 – Bypass LPF		0b	RW
01 :00		ibgclksource[1:0]: 00 = DS bit stream 01 = reserve 10 = reserve 11 = internal bandgap clock		11b	RW

17.2.1.44 Offset F0h: Reserved

17.2.1.45 Offset F8h: MANID—Manufacturing/Process

Table 17-45. Offset F8h: MANID—Manufacturing/Process

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:31:6	**Offset Start:** F8h **Offset End:** FBh	
Size: 32 bit	**Default:** 00000F00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved		00h	
23 :16		Stepping ID — This field is incremented for each stepping of the part. Note: This field can be used by software to differentiate steppings when the Revision ID may not change. Implementation Note: A single Stepping ID can be implemented that is readable from all functions in the chip because all of them are incremented in lock-step.		00h	RO
15 :00	Reserved	Reserved		0F00h	

17.2.2 PCH Thermal Reporting

The Thermal Reporting Registers are located in memory space. The base memory address for the registers is specified at TBARB (B0:D31:F6: Offset 40h - 43h). The individual registers are accessible at TBARB + Offset.

There are two Thermal Sensors. Registers for Thermal Sensor 0 (TS0) starts at offset 00h and Thermal Sensor 1 (TS1) starts at offset 40h

The Manageability Engine (ME) has both read and write access to the thermal sensor offsets 00h-FFh. See the ME C-spec for additional details.

The register layout follows with only the lower 9 bits of the offset listed.

In the table the new registers have the access requirements for host and ME listed. The first provides host access capabilities, and the second is ME access.

17.2.2.1 DIMM Thermal Reporting Configurations

The PCH collects DIMM thermal data via the PECI or Host SMBus depending on the type of CPU on the platform.

Platforms based on CPUs with integrated DIMM SMBuses, the PCH collects DIMM thermal data via the PECI bus. An example CPU with integrated DIMM SMBuses is Intel® Xeon® Processor E5-2600 family.

Platforms based on CPUs without integrated DIMM SMBuses, the PCH collects DIMM thermal data via the platform Host SMBus. An example CPU without integrated DIMM SMBuses is Intel® Xeon® and Intel® Core™ Processors For Communications Infrastructure.

In this document, the following references shall be used:

- Platforms based on CPUs with integrated DIMM SMBuses shall be referred to as **PECI-to-DIMM Processor (PDP)** based platform since the PCH uses the PECI bus to access DIMM thermal data.
- Platforms based on CPUs without integrated DIMM SMBuses shall be referred to as **SMBus-to-DIMM Processor** (**SDP**) based platform since the PCH uses the platform internal Host SMBus to access DIMM thermal data.

Note: The PCH ME FW determines the type of platform and the DIMM access port via a Soft-Strap setting in the SPI Descriptor.

- Refer to *SMBus-to-PECI Bridge Protocol Application Note (Doc# 460415)* for more information.

17.2.3 Thermal Reporting Registers

Table 17-46. Bus 0, Device 31, Function 6: Summary of Thermal Reporting Registers Mapped Through TBARB BAR (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	00h	"Offset 00h: TS0IU—Thermal Sensor 0 In Use" on page 806	00h
01h	01h	"Offset 01h: TS0C—Thermal Sensor 0 Control" on page 807	00h
02h	02h	"Offset 02h: TS0S—Thermal Sensor 0 Status" on page 809	00h
03h	03h	"Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read" on page 810	XXh
04h	07h	"Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point" on page 810	00000000h

Table 17-46. Bus 0, Device 31, Function 6: Summary of Thermal Reporting Registers Mapped Through TBARB BAR (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
08h	08h	"Offset 08h: TS0CO—Thermal Sensor 0 Calibration Offset" on page 812	00h
0Bh	0Bh	"Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd" on page 812	00h
0Ch	0Ch	"Offset 0Ch: TS0ES—Thermal Sensor 0 Error Status" on page 813	00h
0Dh	0Dh	"Offset 0Dh: TS0GPEN—Thermal Sensor 0 General Purpose Event Enables" on page 814	00h
0Eh	0Eh	"Offset 0Eh: TS0PC—Thermal Sensor 0 Policy Control" on page 815	00h
10h	11h	"Offset 10h: CPEC—CPU Power Error Correction Data" on page 816	00h
12h	13h	"Offset 12h: C0TA—CPU0 Temperature Adjust" on page 816	00h
16h	17h	"Offset 16h: C1TA—CPU1Temperature Adjust " on page 817	0000h
1Ah	1Bh	"Offset 1Ah: MC—MPC Control" on page 817	0000h
22h	23h	"Offset 22h: Timestamp" on page 818	0000h
24h	27h	"Offset 24h: DIMMEN—DIMM Enable" on page 819	0000h
30h	31h	"Offset 30h: C0TV—CPU0 Temperature Value" on page 820	0000h
32h	33h	"Offset 32h: C1TV—CPU1 Temperature Value" on page 821	0000h
34h	37h	"Offset 34h: C0EV—CPU0 Energy Value" on page 821	00000000h
38h	3Bh	"Offset 38h: C1EV—CPU1 Energy Value" on page 821	00000000h
3Fh	3Fh	"Offset 3Fh: AE—Alert Enable" on page 822	00h
40h	40h	"Offset 40h: TS1IU—Thermal Sensor 1 In Use" on page 823	00h
41h	41h	"Offset 41h: TS1C—Thermal Sensor 1 Control" on page 824	00h
42h	42h	"Offset 42h: TS1S—Thermal Sensor 1 Status" on page 825	00h
43h	43h	"Offset 43h: TS1TR—Thermal Sensor 1 Thermometer Read" on page 826	XXh
44h	47h	"Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point" on page 826	00000000h
48h	48h	"Offset 48h: TS1CO—Thermal Sensor 1 Calibration Offset" on page 828	00h
4Bh	4Bh	"Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd" on page 829	00h
4Ch	4Ch	"Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status" on page 829	00h
4Dh	4Dh	"Offset 4Dh: TS1GPEN—Thermal Sensor 1 General Purpose Event Enables" on page 831	00h
4Eh	4Eh	"Offset 4Eh: TS1PC—Thermal Sensor 1 Policy Control" on page 832	00h
50h	55h	"Offset 50h: HTS—HOST Turbo Status " on page 833	000000000000h
56h	57h	"Offset 56h: MTL—MCP Temperature Limit" on page 833	0000h
58h	5Fh	"Offset 58h: MTV—MCH Temperature Value" on page 834	0000000000000000h
60h	61h	"Offset 60h: MCPTV—MCP Temperature Value" on page 834	0000h
64h	65h	"Offset 64h: MMPC—Max MCH Power Clamp" on page 834	0000h
66h	67h	"Offset 66h: MMCPPC—Max MCP Power Clamp" on page 835	0000h
82h	82h	"Offset 82h: TS0PIEN—Thermal Sensor 0 PCI Interrupt Event Enables" on page 835	00h
83h	83h	"Offset 83h: TS0LOCK—Thermal Sensor 0 Register Lock Controls" on page 836	00h
98h	9Bh	"Offset 98h: STS—SMBus Turbo Status" on page 836	00000000h
9Ch	9Ch	"Offset 9Ch: SEC—SMBus Event Clear" on page 837	00h

Table 17-46. Bus 0, Device 31, Function 6: Summary of Thermal Reporting Registers Mapped Through TBARB BAR (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
A4h	A7h	"Offset A4h: TC3—Thermal Compares 3" on page 837	00000000h
A8h	ABh	"Offset A8h: TC1—Thermal Compares 1" on page 838	00000000h
ACh	AFh	"Offset ACh: TC2—Thermal Compares 2" on page 838	00000000h
B0h	B3h	"Offset B0h: DIMM0—CPU0 DIMM Data" on page 839	00000000h
B4h	B7h	"Offset B4h: DIMM1—CPU1 DIMM Data" on page 841	00000000h
B8h	BBh	"Offset B8h: DIMMID—DIMM ID" on page 842	00000000h
BCh	BFh	"Offset BCh: ECPCLAMP—EC Power Clamp Data" on page 844	00000000h
C2h	C2h	"Offset C2h: TS1PIEN—Thermal Sensor 1PCI Interrupt Event Enables" on page 844	00h
C3h	C3h	"Offset C3h: TS1LOCK—Thermal Sensor 1 Register Lock Controls" on page 845	00h
D8h	DBh	"Offset D8h: ITV—Internal Temperature Values" on page 845	00000000h

Many of the following registers are implemented twice, once for each thermal sensor.

17.2.3.1 Offset 00h: TS0IU—Thermal Sensor 0 In Use

Table 17-47. Offset 00h: TS0IU—Thermal Sensor 0 In Use

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 00h **Offset End:** 00h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved		00h	
00	TS0IU	Thermal Sensor In Use — Software semaphore bit. After a core-well reset, a read to this bit returns a 0. After the first read, subsequent reads will return a 1. A write of a 1 to this bit will reset the next read value to 0. Writing a 0 to this bit has no effect. Software can poll this bit until it reads a 0, and will then own the usage of the thermal sensor. This bit has no other effect on the hardware, and is only used as a semaphore among various independent software threads that may need to use the thermal sensor. Software that reads this register but does not intend to claim exclusive access of the thermal sensor must write a one to this bit if it reads a 0, in order to allow other software threads to claim it. NOTE: The ME only reads the thermal interface registers and as such ME reads to TSIU will not set the bit.		0b	RS/WC

17.2.3.2 Offset 01h: TS0C—Thermal Sensor 0 Control

This register controls the operation of the thermal sensor.

Table 17-48. Offset 01h: TS0C—Thermal Sensor 0 Control (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 01h **Offset End:** 01h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	TS0E	Thermal Sensor enable — This bit enables thermal reporting. Lockable via TSCO bit 7: 0 -Disabled 1 -Enabled		0b	RWL
06	Reserved	Reserved		0b	RW
05 :04	DHA0	Digital Hysteresis Amount: This bit determines whether no offset, 1.0°C, 2.0°C, or 3.0°C is used for hysteresis for the trip points. 00 = digital hysteresis disabled 01 = enabled, offset is ~1.0°C 10 = enabled, offset is ~2.0°C 11 = enabled, offset is ~3.0°C When set to non-zero value, the hysteresis works as follows: Assume a Aux trip point of 100°C. On the cold-to-hot case, where the device is heating up, the Aux trip will occur at 100°C. Later as the device cools down, the hot-to-cool trip will occur at 97°C, assuming a setting of 11. So the hysteresis only affects the hot-to-cool transition. This prevents thrashing on interrupts as the part hovers around 100°C. This applies to Aux1, Aux2, Hot, and Cat trip points. ***Note:*** Not used in PCH		00b	RW

Table 17-48. Offset 01h: TS0C—Thermal Sensor 0 Control (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 01h **Offset End:** 01h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
03 :02	SE0	Sequencer enable and Rate — These bits enable the thermal sensor to sequence between four measurement functions and set the sequencer rate. When the sequencer is disabled, the Thermal Sensor Temperature Trip Point Setting [Catastrophic Trip point setting] values will be routed to bank A of the DAC and TSCO 6 lsbs is routed to bank B of the DAC. The Catastrophic trip is fully functional and the Hot trip is functional at fixed offset of ~2.0°C below the catastrophic. The Auxiliary trip and HW thermometer are not functional in this mode. When sequencer is enabled, the Catastrophic, Hot, Auxiliary and thermometer circuits will all operate using the programmed trip points and sequencer rate. Note: When disabling the sequencer while thermometer running, the sequencer will finish the current cycle until it starts another one. Lockable via TSCO bit 7. 00: sequencer disabled (i.e, analog sensor mode) 01: enabled, eight clock mode (for testing digital logic) 10: enabled,104 clock mode (normal sequencer operation), provides 4.16 uS settling time @ 25 MHz 11: enabled,820 clock mode (fall back sequencer option), provides 32.8 uS settling time @ 25 MHz. ***Note:*** Sequencer is not supported in PCH and has been permenently disabled in this register.		00b	RO
01	TS0OS	Thermal Sensor output select: This bit muxes between the two comparator outputs. Normally Catastrophic is used. Lockable via TSCO bit 7. 0 = Catastrophic 1 = Hot ***Note:*** Not used in PCH		0b	RWL
00	Reserved	Reserved		0b	

17.2.3.3 Offset 02h: TS0S—Thermal Sensor 0 Status

This read only register provides trip point and other status of the thermal sensor.

Table 17-49. Offset 02h: TS0S—Thermal Sensor 0 Status

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 02h **Offset End:** 02h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	CTI0	Catastrophic Trip Indicator — a 1 indicates that the temperature is above the catastrophic setting.		0b	RO
06	HTI0	Hot Trip Indicator — a 1 indicates that the temperature is above the Hot setting.		0b	RO
05	ATI0	Auxiliary Trip Indicator — a 1 indicates that the temperature is above the Auxiliary setting.		0b	RO
04	Reserved	Reserved		0b	
03	ATI0	Auxiliary2 Trip Indicator — a 1 indicates that the temperature is above the Auxiliary2 setting.		0b	RO
02	Reserved	Reserved		0b	
01	DCCR0	Direct Catastrophic Comparator Read — This bit reads the output of the Catastrophic comparator directly, without latching via the sequencer circuit. Used for testing.		0b	RO
00	DHCR0	Direct Hot comparator read — This bit reads the output of the Hot comparator directly, without latching via the sequencer circuit. Used for testing.		0b	RO

17.2.3.4 Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read

Table 17-50. Offset 03h: TS0TR—Thermal Sensor 0 Thermometer Read

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 03h **Offset End:** 03h	
Size: 8 bit	**Default:**	XXh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TR0	Temperature reading, where calculated value is (TR[7:0] – 50 deg C). These register bits reflect "Offset A8h: CCTS0TL1—Thermal Sensor 0 Top Logic 1" on page 797 register 'otstemperature' bits [8:1]. Default value is X it depends on temperature at power on.			XXh	RO

17.2.3.5 Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point

This register sets thermal trip points.

This register should be set with setting of "Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd" on page 812

Table 17-51. Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point (Sheet 1 of 2)

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:**	00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	A2TPS0	AUX2 Trip point setting — When temperature reaches the value indicated by this bus, alert signal ("Offset 02h: TS0S—Thermal Sensor 0 Status" on page 809) bit 03 is asserted. • Default value has been set for 130°C • Calculated value is (A2TPS[7:0] - 50°C) ***Note:*** This register field provides an optional temperature trip pont. By default this trip point has been disabled by setting it to a value greater than the catastrophic trip point to prevent it from unitentionaly firing. If required this field must be programmed to an appropriate value before using in normal operation. • AUX2 has a built in hysteresis of 2°C. Care must be taken not to program the bits in this field to a value less than 2 or wraparound will occur on the final value used. These bits are lockable via programming the policy-lock down bit (bit 7) of TSPC reg			B4h	RWL

Table 17-51. Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point (Sheet 2 of 2)

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:**	00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16	ATPS0	AUX Trip point setting — When temperature reaches the value indicated by this bus, alert signal ("Offset 02h: TS0S—Thermal Sensor 0 Status" on page 809) bit 05 is asserted. • Default value has been set for 130°C • Calculated value is (ATPS[7:0] - 50°C) ***Note:*** This register field provides an optional temperature trip pont. By default this trip point has been disabled by setting it to a value greater than the catastrophic trip point to prevent it from unitentionaly firing. If required this field must be programmed to an appropriate value before using in normal operation. • AUX has a built in hysteresis of 2°C. Care must be taken not to program the bits in this field to a value less than 2 or wraparound will occur on the final value used. These bits are lockable via programming the policy-lock down bit (bit 7) of TSPC reg			B4h	RWL
15 :08	HTPS0	itsalertthreshon[7:0] — (Hot On value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned on. • Default value has been set for 105°C • Calculated value is (itsalertthreshon[7:0] - 50°C) • Lockable via TSLOCK bit 2.			9Bh	RWL
07 :00	CTPS0	itstripthreshon[7:0] — (Catastrophic value) When temperature reaches the value indicated by this bus, trip signal (otstemperaturetrip) is turned on. • Default value has been set for 112°C • Calculated value is (itstripthreshon[7:0] - 50°C) • Lockable via TSCO bit 7.			A2h	RWL

17.2.3.6 Offset 08h: TS0CO—Thermal Sensor 0 Calibration Offset

Table 17-52. Offset 08h: TS0CO—Thermal Sensor 0 Calibration Offset

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:**	B:31:6	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	LBC0	Lock bit for Catastrophic — This bit, when written to a 1, locks the Catastrophic programming interface, including bits 6:0 of this register and bits 7:0 of TSTTP, bits 1,2, 3 and 7 of TSC, and bits 0,7 of TSTM. This bit may only be set to a 0 by a Host Partitioned reset (bb_rst). Writing a 0 to this bit has no effect. *CF9 warm reset is a Host Partitioned Reset.*			0b	RWLO
06 :00	CO0	Calibration Offset — This field contains the current calibration offset for the Thermal Sensor DAC inputs. The calibration offset is a twos complement signed number which is added to the catastrophic/hot/aux/temp value before going to the DAC. This field is Read/Write and can be modified by Software unless locked by setting bit 7 of this register. While this is a seven-bit field, the 7th bit is sign extended to 9 bits for TCO operation. The range of 00h to 3fh corresponds to 0 0000 0000 to 0 0011 1111. The range of 41h to 7fh corresponds to 1 1100 0001 (i.e, negative 3fh) to 1 1111 1111 (i.e, negative 1), respectively. ***Note:*** Not used in PCH			00h	RWL

17.2.3.7 Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd

This register sets Hot off thermal trip point.

This register should be set with setting of "Offset 04h: TS0TTP—Thermal Sensor 0 Temperature Trip Point" on page 810

Table 17-53. Offset 0Bh: TS0TTPC—Thermal Sensor 0 Temperature Trip Point Cont'd

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:**	B:31:6	**Offset Start:** 0Bh **Offset End:** 0Bh	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HOTP0	itsalertthreshoff[7:0] — (Hot off value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned off. • Default value has been set for 103°C • Calculated value is (itsalertthreshoff [7:0] - 50°C) Lockable via TSLOCK bit 2.			99h	RWL

17.2.3.8 Offset 0Ch: TS0ES—Thermal Sensor 0 Error Status

Table 17-54. Offset 0Ch: TS0ES—Thermal Sensor 0 Error Status

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	AUX2HL0	Aux2 High-to-Low Event: 1 = Indicates that an Aux2 Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
06	CATHL0	Catastrophic High-to-Low Event: 1 = Indicates that a Catastrophic Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit. NOTE: This will never be seen. Power cycle will clear this bit.			0b	RWC
05	HOTHL0	Hot High-to-Low Event: 1 = Indicates that a Hot Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
04	AUXHL0	Aux High-to-Low Event: 1 = Indicates that an Aux Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
03	AUX2LH0	Aux2 Low-to-High Event: 1 = Indicates that an Aux2 Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
02	CATLH0	Catastrophic Low-to-High Event: 1 = Indicates that a Catastrophic Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
01	HOTLH0	Hot Low-to-High Event: 1 = Indicates that a Hot Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 -No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
00	AUXLH0	Aux Low-to-High Event: 1 = Indicates that an Aux Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC

17.2.3.9 Offset 0Dh: TS0GPEN—Thermal Sensor 0 General Purpose Event Enables

This register controls the conditions that result in the General Purpose Event (GPE) signal from the Thermal Sensor (TS) logic to assert. When the TS GPE signal asserts, the GPE block reports a '1' in the TCOSCI_STS bit. The MCH thermal sensor logic also sets this bit.

Table 17-55. Offset 0Dh: TS0GPEN—Thermal Sensor 0 General Purpose Event Enables

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 0Dh **Offset End:** 0Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Aux2 High-to-Low Enable — See the description for bit 6.		0b	RW
06		Catastrophic High-to-Low Enable: When set to 1, the thermal sensor logic asserts its General Purpose Event signal to the GPE block when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in the GPE signal assertion.		0b	RW
05		Hot High-to-Low Enable — See the description for bit 6.		0b	RW
04		Aux High-to-Low Enable — See the description for bit 6.		0b	RW
03		Aux2 Low-to-High Event — See the description for bit 6.		0b	RW
02		Catastrophic Low-to-High Enable — See the description for bit 6.		0b	RW
01		Hot Low-to-High Event — See the description for bit 6.		0b	RW
00		Aux Low-to-High Event — See the description for bit 6.		0b	RW

17.2.3.10 Offset 0Eh: TS0PC—Thermal Sensor 0 Policy Control

This register enables specific events to generate an SMI#.

Table 17-56. Offset 0Eh: TS0PC—Thermal Sensor 0 Policy Control

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07		Policy Lock-Down Bit — When written to 1, this bit prevents any more writes to this register (offset 0Eh) and bits 31:16 of the Thermal Sensor Temperature Trip Point register (offset 04h). The "Lock Bit for Catastrophic" (bit 7 of offset 08h) and the "Lock Bit for Hot" (bit 2 of offset 83h) must also be 1 when this bit is set to 1. When 0, the thermal policies can be programmed and modified. NOTE: For the second thermal sensor, the registers are located 40h above the offsets for the first thermal sensor. This bit is reset by a Host Partitioned Reset (bb_rst). *CF9 warm reset is a Host Partitioned Reset.*			0b	RWLO
06		Catastrophic Power-Down Enable — When set to 1, the power management logic transitions to the S5 state when a catastrophic temperature is detected by the sensor. The transition to the S5 state must be unconditional (like the Power Button Override Function). The assertion of the signal controlling the main power well (SLP_S3# or SLP_S4#) must occur within 100 µsec of the detection of the catastrophic condition. NOTE: The thermal sensor and response logic is in the core/main power well; therefore, detection of a catastrophic temperature is limited to times when this well is powered and out of reset. NOTE: The sequencer mode can delay the detection of a catastrophic temperature. Assuming the 512-clock sequencer mode is in use, a catastrophic condition may not be detected internally for up to 16.4 µsecs. Therefore the total delay from the catastrophic condition to the assertion of the SLP_Sx# signals can be 116.4 µsecs.			0	RW
05 :04	Reserved	Reserved			00b	
03		SMI Enable on Aux2 Thermal Sensor Trip: 1: Enables SMI# assertions on Aux2 thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for Aux2 thermal events			0b	RW
02		SMI Enable on Catastrophic Thermal Sensor Trip: 1: Enables SMI# assertions on catastrophic thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for catastrophic thermal events.			0b	RW
01		SMI Enable on Hot Thermal Sensor Trip: 1: Enables SMI# assertions on hot thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for hot thermal events			0b	RW
00		SMI Enable on Aux Thermal Sensor Trip: 1: Enables SMI# assertions on aux thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for aux thermal events			0b	RW

17.2.3.11 Offset 10h: CPEC—CPU Power Error Correction Data

Table 17-57. Offset 10h: CPEC—CPU Power Error Correction Data

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 10h **Offset End:** 11h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		CPU Power Error Correction Data — This register is RW to host and has no functionality in PCH. Locked by CTC[7]		00h	RW

17.2.3.12 Offset 12h: C0TA—CPU0 Temperature Adjust

Table 17-58. Offset 12h: C0TA—CPU0 Temperature Adjust

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 12h **Offset End:** 13h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Host writes the CPU0's TJmax into this register. ME uses the value to create the CPU0's absolute temperature. The value received from the cpu over PECI is a negative offset relative to the C0TA value. Locked by CTC[7].		00h	RW

17.2.3.13 Offset 16h: C1TA—CPU1 Temperature Adjust

Table 17-59. Offset 16h: C1TA—CPU1Temperature Adjust

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 16h **Offset End:** 17h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Host writes the CPU1's TJmax into this register. ME uses the value to create the CPU1's absolute temperature. The value received from the cpu over PECI is a negative offset relative to the C1TA value. Locked by CTC[7].		00h	RW

17.2.3.14 Offset 1Ah: MC—MPC Control

All the bits below are used by ME FW to enable different thermal reporting features.

BIOS: Each OEM will program the following differently, based on that OEM's configuration.

Table 17-60. Offset 1Ah: MC—MPC Control (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 1Ah **Offset End:** 1Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15		**2nd CPU Present:** 1 = 2nd CPU is present in the system. BIOS sets this bit to '1' in DP systems. When the 2nd CPU is present, all PECI Thermal Reporting commands are executed to both CPUs. 0 = 2nd CPU **NOT** present. BIOS sets this to '0' in UP systems		0b	RW
14 :13	Reserved	Reserved		0b	RW
12		System Thermal Reporting Enable: 1 = Enable System Thermal Reporting 0 = Disable System Thermal Reporting		0b	RW
11 :08	Reserved	Reserved		0b	RW
07		CPU Temperature Read Enable: 1 = Enable PECI Reads of CPU Temp 0 = Disable PECI Reads of CPU Temp		0b	RW

Table 17-60. Offset 1Ah: MC—MPC Control (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 1Ah **Offset End:** 1Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06		CPU Energy Read Enable 1 = Enable PECI Reads of CPU Energy 0 = Disable PECI Reads of CPU Energy.		0b	RW
05		PCH Temperature Read Enable 1 = Enable PCH Temperature Read 0 = Disable PCH Temperature Read.		0b	RW
04 :00	Reserved	Reserved		0b	RW

17.2.3.15 Offset 22h: Timestamp

Table 17-61. Offset 22h: Timestamp

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 22h **Offset End:** 23h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15		When ME writes a 1 to this bit, bits [11:0] are loaded from the free running 10 msec counter.		0b	RO
14 :12	Reserved	Reserved		00b	
11 :00		Timestamp — This register provides a RO count of 10 msec pulses for timestamp purposes. SW can read the data and the timestamp and wait several seconds and read the two again. It takes the difference between the 2 samples to calculate power over a large window. SW is responsible for handling counter rollover, which occurs every 40 seconds. NOTE: Between any 2 samples, the difference with respect to the data collection could be off by 1 clock, since the timing on PECI is not precise with the timestamp itself.		000h	RO

17.2.3.16 Offset 24h: DIMMEN—DIMM Enable

The bits are configured by BIOS and used by ME FW to identify DIMMs that are present in the system and can be enabled for thermal reporting. This register is used PDP and SDP applications.

For SDP platforms, Bits[1:0] are used to enable two DIMMs in CH0; and Bits[5:4] are for two DIMMs in CH1.

BIOS sets the bits for the DIMMs that are physically present and wants to enable for thermal polling.

Table 17-62. Offset 24h: DIMMEN—DIMM Enable (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28		**CPU1/CH3 DIMM_Enable** Enables reads of DIMM's populated in CPU1/CH3 and have an onboard TS. Bit[28] = CPU1/CH3/DIMM0 Bit[29] = CPU1/CH3/DIMM1 Bit[30] = CPU1/CH3/DIMM2 Bit[31] = Reserved		0b	RW
27 :24		**CPU1/CH2 DIMM_Enable** Enables reads of DIMM's populated in CPU1/CH2 and have an onboard TS. Bit[24] = CPU1/CH2/DIMM0 Bit[25] = CPU1/CH2/DIMM1 Bit[26] = CPU1/CH2/DIMM2 Bit[27] = Reserved		0b	RW
23 :20		**CPU1/CH1 DIMM_Enable** Enables reads of DIMM's populated in CPU1/CH1 and have an onboard TS. Bit[20] = CPU1/CH1/DIMM0 Bit[21] = CPU1/CH1/DIMM1 Bit[22] = CPU1/CH1/DIMM2 Bit[23] = Reserved		0b	RW
19 :16		**CPU1/CH0 DIMM_Enable** Enables reads of DIMM's populated in CPU1/CH0 and have an onboard TS. Bit[16] = CPU1/CH0/DIMM0 Bit[17] = CPU1/CH0/DIMM1 Bit[18] = CPU1/CH0/DIMM2 Bit[19] = Reserved		0b	RW
15 :12		**CPU0/CH3 DIMM_Enable** Enables reads of DIMM's populated in CPU0/CH3 and have an onboard TS. Bit[12] = CPU0/CH3/DIMM0 Bit[13] = CPU0/CH3/DIMM1 Bit[14] = CPU0/CH3/DIMM2 Bit[15] = Reserved		0b	RW

Table 17-62. Offset 24h: DIMMEN—DIMM Enable (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11 :08		**CPU0/CH2 DIMM_Enable** Enables reads of DIMM's populated in CPU0/CH2 and have an onboard TS. Bit[8] = CPU0/CH2/DIMM0 Bit[9] = CPU0/CH2/DIMM1 Bit[10] = CPU0/CH2/DIMM2 Bit[11] = Reserved		0b	RW
07 :04		**CPU0/CH1 DIMM_Enable** Enables reads of DIMM's populated in CPU0/CH1 and have an onboard TS. Bit[4] = CPU0/CH1/DIMM0 Bit[5] = CPU0/CH1/DIMM1 Bit[6] = CPU0/CH1/DIMM2 Bit[7] = Reserved ***Note:*** **Bit[4]** and **Bit[5]** are used to enable two DIMMs in **CH1** on SDP platforms		0b	RW
03 :00		**CPU0/CH0 DIMM_Enable** Enables reads of DIMM's populated in CPU0/CH0 and have an onboard TS. Bit[0] = CPU0/CH0/DIMM0 Bit[1] = CPU0/CH0/DIMM1 Bit[2] = CPU0/CH0/DIMM2 Bit[3] = Reserved ***Note:*** **Bit[0]** and **Bit[1]** are used to enable two DIMMs in **CH0** on SDP platforms		0b	RW

17.2.3.17 Offset 30h: C0TV—CPU0 Temperature Value

Table 17-63. Offset 30h: C0TV—CPU0 Temperature Value

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 30h **Offset End:** 31h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		CPU0 Temperature — This provides the CPU0 temperature. For format, See the spec for EC support. This is byte 3 of the EC data.		0000h	RO

17.2.3.18 Offset 32h: C1TV—CPU1 Temperature Value

Table 17-64. Offset 32h: C1TV—CPU1 Temperature Value

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 32h **Offset End:** 33h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		CPU1 Temperature — This provides the CP1 temperature. for format, See the spec for EC support. This is byte 4 of the EC data.			RO

17.2.3.19 Offset 34h: C0EV—CPU0 Energy Value

Table 17-65. Offset 34h: C0EV—CPU0 Energy Value

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 34h **Offset End:** 37h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CPU0 Power — ME writes the CPU0 power here after collecting it over PECI. This is bytes 14:11 of the EC data. Byte 11 of the EC data is bits [7:0] of this register.		00000000h	RO

17.2.3.20 Offset 38h: C1EV—CPU1 Energy Value

Table 17-66. Offset 38h: C1EV—CPU1 Energy Value

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 38h **Offset End:** 3Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00		CPU1 Power: ME writes the CPU1 power here after collecting it over PECI. This register is used for CPU1 in a DP system. It is not used in a uP system.		00000000h	RO

17.2.3.21 Offset 3Fh: AE—Alert Enable

BIOS: Some OEM's may enable the alert feature, whereby the PCH indicates through a pin if a device is outside the thermal bounds. In general BIOS should not program the following register at all. If the OEM wishes to enable the pin-based alert, then BIOS should write a '1' to the bits below that should be enabled for the particular Embedded Controller on that platform.

Table 17-67. Offset 3Fh: AE—Alert Enable

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 3Fh **Offset End:** 3Fh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Lock Enable — When '1', this bit (Lock Enable) and CTC[6:0] are locked and the following registers are locked against any writes from the host: -CPU Power Error Correction 0x10 -C0TA 0x12 -PTA 0x14 -C1TA 0x16 This bit is reset by Host Partitioned Reset (bb_rst). *CF9 warm reset is a Host Partitioned Reset.*		0b	RWL
06		CPU Alert Enable - Locked by CTC[7]		0b	RWL
05		MCH Alert Enable - Locked by CTC[7]		0b	RWL
04		PCH Alert Enable - Locked by CTC[7]		0b	RWL
03		DIMM Alert Enable — the actual DIMMs that are read and used for comparison are enabled in a different register - Locked by CTC[7]		0b	RWL
02 :00	Reserved	Reserved - Locked by CTC[7]		000b	RWL

17.2.3.22 Offset 40h: TS1IU—Thermal Sensor 1 In Use

Table 17-68. Offset 40h: TS1IU—Thermal Sensor 1 In Use

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 40h **Offset End:** 40h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved		00h	
00	TS1IU	Thermal Sensor In Use — Software semaphore bit. After a core-well reset, a read to this bit returns a 0. After the first read, subsequent reads will return a 1. A write of a 1 to this bit will reset the next read value to 0. Writing a 0 to this bit has no effect. Software can poll this bit until it reads a 0, and will then own the usage of the thermal sensor. This bit has no other effect on the hardware, and is only used as a semaphore among various independent software threads that may need to use the thermal sensor. Software that reads this register but does not intend to claim exclusive access of the thermal sensor must write a one to this bit if it reads a 0, in order to allow other software threads to claim it. NOTE: The ME only reads the thermal interface registers and as such ME reads to TSIU will not set the bit.		0b	RS/WC

17.2.3.23 Offset 41h: TS1C—Thermal Sensor 1 Control

This register controls the operation of the thermal sensor.

Table 17-69. Offset 41h: TS1C—Thermal Sensor 1 Control

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6		**Offset Start:** 41h **Offset End:** 41h	
Size: 8 bit	**Default:** 00h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	TS1E	Thermal Sensor enable — This bit enables thermal reporting. Lockable via TSCO bit 7: 0 -Disabled 1 -Enabled			0b	RWL
06	Reserved	Reserved			0b	RW
05 :04	DHA1	Digital Hysteresis Amount: This bit determines whether no offset, 1.0°C, 2.0°C, or 3.0°C is used for hysteresis for the trip points. 00 = digital hysteresis disabled 01 = enabled, offset is ~1.0°C 10 = enabled, offset is ~2.0°C 11 = enabled, offset is ~3.0°C When set to non-zero value, the hysteresis works as follows: Assume a Aux trip point of 100°C. On the cold-to-hot case, where the device is heating up, the Aux trip will occur at 100°C. Later as the device cools down, the hot-to-cool trip will occur at 97°C, assuming a setting of 11. So the hysteresis only affects the hot-to-cool transition. This prevents thrashing on interrupts as the part hovers around 100°C. This applies to Aux1, Aux2, Hot, and Cat trip points. ***Note:*** Not used in PCH			00b	RW
03 :02	SE1	Sequencer enable and Rate — These bits enable the thermal sensor to sequence between four measurement functions and set the sequencer rate. When the sequencer is disabled, the Thermal Sensor Temperature Trip Point Setting [Catastrophic Trip point setting] values will be routed to bank A of the DAC and TSCO 6 lsbs is routed to bank B of the DAC. The Catastrophic trip is fully functional and the Hot trip is functional at fixed offset of ~2.0°C below the catastrophic. The Auxiliary trip and HW thermometer are not functional in this mode. When sequencer is enabled, the Catastrophic, Hot, Auxiliary and thermometer circuits will all operate using the programmed trip points and sequencer rate. Note: When disabling the sequencer while thermometer running, the sequencer will finish the current cycle until it starts another one. Lockable via TSCO bit 7. 00: sequencer disabled (i.e, analog sensor mode) 01: enabled, eight clock mode (for testing digital logic) 10: enabled,104 clock mode (normal sequencer operation), provides 4.16 uS settling time @ 25 MHz 11: enabled,820 clock mode (fall back sequencer option), provides 32.8 uS settling time @ 25 MHz. ***Note:*** Sequencer is not supported in the PCH and has been permanently disabled in this register.			00b	RO
01	TS1OS	Thermal Sensor output select: This bit muxes between the two comparator outputs. Normally Catastrophic is used. Lockable via TSCO bit 7. 0 = Catastrophic 1 = Hot ***Note:*** Not used in the PCH			0b	RWL
00	Reserved	Reserved			0b	

17.2.3.24 Offset 42h: TS1S—Thermal Sensor 1 Status

This read only register provides trip point and other status of the thermal sensor.

Table 17-70. Offset 42h: TS1S—Thermal Sensor 1 Status

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 42h **Offset End:** 42h	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	CTI1	Catastrophic Trip Indicator — a 1 indicates that the temperature is above the catastrophic setting.			0b	RO
06	HTI1	Hot Trip Indicator — a 1 indicates that the temperature is above the Hot setting.			0b	RO
05	ATI1	Auxiliary Trip Indicator — a 1 indicates that the temperature is above the Auxiliary setting.			0b	RO
04	Reserved	Reserved			0b	
03	ATI1	Auxiliary2 Trip Indicator — a 1 indicates that the temperature is above the Auxiliary2 setting.			0b	RO
02	Reserved	Reserved			0b	
01	DCCR1	Direct Catastrophic Comparator Read — This bit reads the output of the Catastrophic comparator directly, without latching via the sequencer circuit. Used for testing.			0b	RO
00	DHCR1	Direct Hot comparator read — This bit reads the output of the Hot comparator directly, without latching via the sequencer circuit. Used for testing.			0b	RO

17.2.3.25 Offset 43h: TS1TR—Thermal Sensor 1 Thermometer Read

Table 17-71. Offset 43h: TS1TR—Thermal Sensor 1 Thermometer Read

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 43h **Offset End:** 43h	
Size: 8 bit	**Default:** XXh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	TR1	Temperature reading, where calculated value is (TR[7:0] – 50 deg C). These register bits reflect "Offset C8h: CCTS1TL1—Thermal Sensor 1 Top Logic 1" on page 801 register 'otstemperature' bits [8:1]. Default value is X it depends on temperature at power on.		XXh	RO

17.2.3.26 Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point

This register sets thermal trip points. This register should be set with setting of "Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd" on page 829

Table 17-72. Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	A2TPS1	AUX2 Trip point setting — When temperature reaches the value indicated by this bus, alert signal ("Offset 42h: TS1S—Thermal Sensor 1 Status" on page 825) bit 03 is asserted. • Default value has been set for 130°C • Calculated value is (A2TPS[7:0] - 50°C) ***Note:*** This register field provides an optional temperature trip pont. By default this trip point has been disabled by setting it to a value greater than the catastrophic trip point to prevent it from unitentionaly firing. If required this field must be programmed to an appropriate value before using in normal operation. • AUX2 has a built in hysteresis of 2°C. Care must be taken not to program the bits in this field to a value less than 2 or wraparound will occur on the final value used. These bits are lockable via programming the policy-lock down bit (bit 7) of TSPC reg		B4h	RWL

Table 17-72. Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point (Sheet 2 of 2)

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:**	00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16	ATPS1	AUX Trip point setting — When temperature reaches the value indicated by this bus, alert signal ("Offset 42h: TS1S—Thermal Sensor 1 Status" on page 825) bit 05 is asserted. • Default value has been set for 130°C • Calculated value is (ATPS[7:0] - 50°C) ***Note:*** This register field provides an optional temperature trip pont. By default this trip point has been disabled by setting it to a value greater than the catastrophic trip point to prevent it from unitentionaly firing. If required this field must be programmed to an appropriate value before using in normal operation. • AUX has a built in hysteresis of 2°C. Care must be taken not to program the bits in this field to a value less than 2 or wraparound will occur on the final value used. These bits are lockable via programming the policy-lock down bit (bit 7) of TSPC reg			B4h	RWL
15 :08	HTPS1	itsalertthreshon[7:0] — (Hot On value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned on. • Default value has been set for 105°C • Calculated value is (itsalertthreshon[7:0] - 50°C) Lockable via TSLOCK bit 2.			9Bh	RWL
07 :00	CTPS1	itstripthreshon[7:0] — (Catastrophic value) When temperature reaches the value indicated by this bus, trip signal (otstemperaturetrip) is turned on. • Default value has been set for 112°C • Calculated value is (itstripthreshon[7:0] - 50°C) Lockable via TSCO bit 7.			A2h	RWL

17.2.3.27 Offset 48h: TS1CO—Thermal Sensor 1 Calibration Offset

Table 17-73. Offset 48h: TS1CO—Thermal Sensor 1 Calibration Offset

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 48h **Offset End:** 48h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07	LBC1	Lock bit for Catastrophic — This bit, when written to a 1, locks the Catastrophic programming interface, including bits 6:0 of this register and bits 7:0 of TSTTP, bits 1,2, 3 and 7 of TSC, and bits 0,7 of TSTM. This bit may only be set to a 0 by a Host Partitioned reset (bb_rst). Writing a 0 to this bit has no effect. *CF9 warm reset is a Host Partitioned Reset.*		0b	RWLO
06 :00	CO1	Calibration Offset — This field contains the current calibration offset for the Thermal Sensor DAC inputs. The calibration offset is a twos complement signed number which is added to the catastrophic/hot/aux/temp value before going to the DAC. This field is Read/Write and can be modified by Software unless locked by setting bit 7 of this register. While this is a seven-bit field, the 7th bit is sign extended to 9 bits for TCO operation. The range of 00h to 3fh corresponds to 0 0000 0000 to 0 0011 1111. The range of 41h to 7fh corresponds to 1 1100 0001 (i.e, negative 3fh) to 1 1111 1111 (i.e, negative 1), respectively. ***Note:*** Not used in the PCH		00h	RWL

17.2.3.28 Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd

This register sets Hot off thermal trip point.

This register should be set with setting of "Offset 44h: TS1TTP—Thermal Sensor 1 Temperature Trip Point" on page 826

Table 17-74. Offset 4Bh: TS1TTPC—Thermal Sensor 1 Temperature Trip Point Cont'd

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:**	B:31:6	**Offset Start:** 4Bh **Offset End:** 4Bh	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HOTP1	itsalertthreshoff[7:0] — (Hot off value) When temperature reaches the value indicated by this bus, alert signal (otstemperaturealrt) is turned off. • Default value has been set for 103°C • Calculated value is (itsalertthreshoff [7:0] - 50°C) Lockable via TSLOCK bit 2.			99h	RWL

17.2.3.29 Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status

Table 17-75. Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status (Sheet 1 of 2)

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:**	B:31:6	**Offset Start:** 4Ch **Offset End:** 4Ch	
Size: 8 bit	**Default:**	00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07	AUX2HL1	Aux2 High-to-Low Event: 1 = Indicates that an Aux2 Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
06	CATHL1	Catastrophic High-to-Low Event: 1 = Indicates that a Catastrophic Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit. NOTE: This will never be seen. Power cycle will clear this bit.			0b	RWC
05	HOTHL1	Hot High-to-Low Event: 1 = Indicates that a Hot Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC

Table 17-75. Offset 4Ch: TS1ES—Thermal Sensor 1 Error Status (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:**	B:31:6	**Offset Start:** 4Ch **Offset End:** 4Ch	
Size: 8 bit	**Default:** 00h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
04	AUXHL1	Aux High-to-Low Event: 1 = Indicates that an Aux Thermal Sensor event based on a higher to lower temperature transition thru the trip point 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
03	AUX2LH1	Aux2 Low-to-High Event: 1 = Indicates that an Aux2 Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
02	CATLH1	Catastrophic Low-to-High Event: 1 = Indicates that a Catastrophic Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
01	HOTLH1	Hot Low-to-High Event: 1 = Indicates that a Hot Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 -No trip for this event Software must write a 1 to clear this status bit.			0b	RWC
00	AUXLH1	Aux Low-to-High Event: 1 = Indicates that an Aux Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0 = No trip for this event Software must write a 1 to clear this status bit.			0b	RWC

17.2.3.30 Offset 4Dh: TS1GPEN—Thermal Sensor 1 General Purpose Event Enables

This register controls the conditions that result in the General Purpose Event (GPE) signal from the Thermal Sensor (TS) logic to assert. When the TS GPE signal asserts, the GPE block reports a '1' in the TCOSCI_STS bit. The MCH thermal sensor logic also sets this bit.

Table 17-76. Offset 4Dh: TS1GPEN—Thermal Sensor 1 General Purpose Event Enables

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 4Dh **Offset End:** 4Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Aux2 High-to-Low Enable — See the description for bit 6.		0b	RW
06		Catastrophic High-to-Low Enable: When set to 1, the thermal sensor logic asserts its General Purpose Event signal to the GPE block when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in the GPE signal assertion.		0b	RW
05		Hot High-to-Low Enable — See the description for bit 6.		0b	RW
04		Aux High-to-Low Enable — See the description for bit 6.		0b	RW
03		Aux2 Low-to-High Event — See the description for bit 6.		0b	RW
02		Catastrophic Low-to-High Enable — See the description for bit 6.		0b	RW
01		Hot Low-to-High Event — See the description for bit 6.		0b	RW
00		Aux Low-to-High Event — See the description for bit 6.		0b	RW

17.2.3.31 Offset 4Eh: TS1PC—Thermal Sensor 1 Policy Control

This register enables specific events to generate an SMI#.

Table 17-77. Offset 4Eh: TS1PC—Thermal Sensor 1 Policy Control

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6		**Offset Start:** 4Eh **Offset End:** 4Eh	
Size: 8 bit	**Default:** 00h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07		Policy Lock-Down Bit — When written to 1, this bit prevents any more writes to this register (offset 0Eh) and bits 31:16 of the Thermal Sensor Temperature Trip Point register (offset 04h). The "Lock Bit for Catastrophic" (bit 7 of offset 08h) and the "Lock Bit for Hot" (bit 2 of offset 83h) must also be 1 when this bit is set to 1. When 0, the thermal policies can be programmed and modified. NOTE: For the second thermal sensor, the registers are located 40h above the offsets for the first thermal sensor. This bit is reset by a Host Partitioned Reset (bb_rst). *CF9 warm reset is a Host Partitioned Reset.*		0b	RWLO
06		Catastrophic Power-Down Enable — When set to 1, the power management logic transitions to the S5 state when a catastrophic temperature is detected by the sensor. The transition to the S5 state must be unconditional (like the Power Button Override Function). The assertion of the signal controlling the main power well (SLP_S3# or SLP_S4#) must occur within 100 µsec of the detection of the catastrophic condition. NOTE: The thermal sensor and response logic is in the core/main power well; therefore, detection of a catastrophic temperature is limited to times when this well is powered and out of reset. NOTE: The sequencer mode can delay the detection of a catastrophic temperature. Assuming the 512-clock sequencer mode is in use, a catastrophic condition may not be detected internally for up to 16.4 µsecs. Therefore the total delay from the catastrophic condition to the assertion of the SLP_Sx# signals can be 116.4 µsecs.		0	RW
05 :04	Reserved	Reserved			00b	
03		SMI Enable on Aux2 Thermal Sensor Trip: 1: Enables SMI# assertions on Aux2 thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for Aux2 thermal events			0b	RW
02		SMI Enable on Catastrophic Thermal Sensor Trip: 1: Enables SMI# assertions on catastrophic thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for catastrophic thermal events.			0b	RW
01		SMI Enable on Hot Thermal Sensor Trip: 1: Enables SMI# assertions on hot thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for hot thermal events			0b	RW
00		SMI Enable on Aux Thermal Sensor Trip: 1: Enables SMI# assertions on aux thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for aux thermal events			0b	RW

17.2.3.32 Offset 50h: HTS—HOST Turbo Status

This register provides the Host view of turbo status. It is used by ME to respond to MCTP messages that request Turbo Status. The SMBus Turbo Status is what an EC can write to. So an EC writes to the SMBus Turbo Status register with its request, which causes an interrupt, and the host updates the Host Turbo Status register so that an EC read of the status gets the Host Turbo Status data and not necessarily what it wrote.

Table 17-78. Offset 50h: HTS—HOST Turbo Status

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 50h **Offset End:** 55h	
Size: 48 bit	**Default:** 000000000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
47 :01		See MPC spec for bit definitions — This represents the data bytes 20:15 provided to the EC when it does a read. Byte 15 is bits [7:0].		000000000000h	RW
00		New Valid Value — Host sets this bit to indicate the data in this register is valid. Cleared on write of '1' to MEC register.		0b	RW

17.2.3.33 Offset 56h: MTL—MCP Temperature Limit

Table 17-79. Offset 56h: MTL—MCP Temperature Limit

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 56h **Offset End:** 57h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		MCH Temperature Limit — programmed by BIOS. Scratchpad register for software. Has no functionality in PCH.		0000h	RO

17.2.3.34 Offset 58h: MTV—MCH Temperature Value

Table 17-80. Offset 58h: MTV—MCH Temperature Value

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 58h **Offset End:** 5Fh	
Size: 64 bit	**Default:** 0000000000000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
63 :00		MCH Temperature Value — ME writes the MCH temperature here after collecting it from the DMI PM_Msg. PM_Msg contains 64 bits of data.		0000000000000000h	RO

17.2.3.35 Offset 60h: MCPTV—MCP Temperature Value

Table 17-81. Offset 60h: MCPTV—MCP Temperature Value

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 60h **Offset End:** 61h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		MCP Temperature Value — ME writes the MCP temperature here after reading PECI for CPU temperature. Bits [7:0] represent the max values of the CPU. This is byte 1 of the EC data. Bits[15:8] are reserved.		0000h	RO

17.2.3.36 Offset 64h: MMPC—Max MCH Power Clamp

Table 17-82. Offset 64h: MMPC—Max MCH Power Clamp

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 64h **Offset End:** 65h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Max MCH Power Clamp — This register is RW to host and has no functionality in PCH.		0000h	RW

17.2.3.37 Offset 66h: MMCPPC—Max MCP Power Clamp

Table 17-83. Offset 66h: MMCPPC—Max MCP Power Clamp

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 66h **Offset End:** 67h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00		Max MMCPPC Power Clamp — This register is RW to host and has no functionality in PCH.		0000h	RW

17.2.3.38 Offset 82h: TS0PIEN—Thermal Sensor 0 PCI Interrupt Event Enables

This register controls the conditions that result in the PCI Interrupt signal from the Thermal Sensor (TS) logic to assert.

Note: There is a separate enable register per sensor.

Table 17-84. Offset 82h: TS0PIEN—Thermal Sensor 0 PCI Interrupt Event Enables

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** 82h **Offset End:** 82h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Aux2 High-to-Low Enable — See the description for bit 6.		0b	RW
06		Catastrophic High-to-Low Enable — When set to 1, the thermal sensor logic asserts the Thermal logic PCI INTx signal when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in PCI INTx.		0b	RW
05		Hot High-to-Low Enable — See the description for bit 6.		0b	RW
04		Aux High-to-Low Enable — See the description for bit 6.		0b	RW
03		Aux2 Low-to-High Event — See the description for bit 6.		0b	RW
02		Catastrophic Low-to-High Enable — See the description for bit 6.		0b	RW
01		Hot Low-to-High Event — See the description for bit 6.		0b	RW
00		Aux Low-to-High Event — See the description for bit 6.		0b	RW

17.2.3.39 Offset 83h: TS0LOCK—Thermal Sensor 0 Register Lock Controls

This register permits BIOS to lock the contents of the Catastrophic Trip controls and Hot Trip controls. The Catastrophic Trip controls are locked with TSCO bit 7. Also, bit 0 contains the TBCNTEN bit that controls Thermal Burden Counter enable/disable.

Table 17-85. Offset 83h: TS0LOCK—Thermal Sensor 0 Register Lock Controls

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6		**Offset Start:** 83h **Offset End:** 83h	
Size: 8 bit	**Default:** 00h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved			0h	RW
02	LBH	Hot Trip Lock — This bit, when written to a 1, locks the Hot Trip programming interface, which is bits 15:8 of TSTTP. This bit may only be set to a 0 by a Host Partitioned Reset (bb_rst) reset. Writing a 0 to this bit has no effect. *CF9 warm reset is a Host Partitioned Reset.*			0b	RWL
01 :00	Reserved	Reserved			00b	RW

17.2.3.40 Offset 98h: STS—SMBus Turbo Status

Anytime ME writes this register an interrupt must be triggered. This register is updated by ME when it receives an MCTP msg on SMBus from the external controller. When ME moves the data from that msg and writes it to this register, an interrupt, if enabled, needs to be signaled. The write will set a bit in the Turbo Interrupt Status register, which then may cause the interrupt.

Table 17-86. Offset 98h: STS—SMBus Turbo Status

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6		**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000000h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01		STS bits [31:1]. Bits [47:32] of STS are in TC1 register. For SW bit usage, See the MPC spec. This is received from the EC when the EC does the Write STS Preferences Command			00000000h	RO
00		New Valid Value. STS[0] ME can set this bit by writing a 1 to it. Writing a 0 has no effect. The bit is cleared when the host writes to the SMBus Event Clear[0].			0b	RO

17.2.3.41 Offset 9Ch: SEC—SMBus Event Clear

Table 17-87. Offset 9Ch: SEC—SMBus Event Clear

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** 9Ch **Offset End:** 9Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :01	Reserved	Reserved		00h	
00		When the Host writes a 1 to this bit, it clears bit [0] of the SMBus Turbo Status Register.		0b	WO

17.2.3.42 Offset A4h: TC3—Thermal Compares 3

Table 17-88. Offset A4h: TC3—Thermal Compares 3

Description:					
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:**	00000000h		**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	CPUTCUL	Thermal Compare Upper Limit: — This is the upper limit used to compare against the CPU's temperature. If the CPU's temperature is greater than this value, then the alert GPIO is asserted. Bits [31:0] of this register are set when the EC does the Write Cpu Temp Limits transaction.		0000h	RO
15 :00	CPUTCLL	Thermal Compare Lower Limit: — This is the lower limit used to compare against the CPU's temperature. If the CPU's temperature is lower than this value, then the alert GPIO is asserted.		0000h	RO

17.2.3.43 Offset A8h: TC1—Thermal Compares 1

Table 17-89. Offset A8h: TC1—Thermal Compares 1

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** A8h **Offset End:** ABh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		STS[47:32] — SMbus Status is a total of 48 bits. This field is bits 47:32. Bits 31:0 are in the STS register. This is received from the EC when the EC does the Write STS Preferences Command		0000h	RO
15 :08	Reserved	Reserved		00h	
07 :00	Reserved	Reserved		00h	

17.2.3.44 Offset ACh: TC2—Thermal Compares 2

Table 17-90. Offset ACh: TC2—Thermal Compares 2

Description:					
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6	**Offset Start:** ACh **Offset End:** AFh	
Size: 32 bit	**Default:**	00000000h		**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	DIMMTCUL	DIMM Thermal Compare Upper Limit: Upper limit used to compare against the DIMM's temperature. If the DIMM's temperature is greater than this value, then the alert GPIO is asserted.		00h	RO
23 :16	DIMMTCLL	DIMM Thermal Compare Lower Limit: Lower limit used to compare against the DIMM's temperature. If the DIMM's temperature is lower than this value, then the alert GPIO is asserted.-Bits [31:16] of this register are set when the EC does the Write DIMM Temp Limits transaction.		00h	RO
15 :08	PCHTCUL	PCH Thermal Compare Upper Limit: Upper limit used to compare against the PCH's temperature. If the PCH's temperature is greater than this value, then the alert GPIO is asserted.		00h	RO
07 :00	PCHTCLL	PCH Thermal Compare Lower Limit: Lower limit used to compare against the PCH's temperature. If the PCH's temperature is lower than this value, then the alert GPIO is asserted.-Bits [15:0] of this register are set when the EC does the Write PCH Temp Limits transaction.		00h	RO

17.2.3.45 Offset B0h: DIMM0—CPU0 DIMM Data

This register is used by ME FW to store the temperature values of DIMMs connected to CPU0.

Table 17-91. Offset B0h: DIMM0—CPU0 DIMM Data (Sheet 1 of 2)

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** B0h **Offset End:** B3h	
Size: 32 bit	**Default:**	00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24		**PDP Platform: CPU0/CH3_DIMM_TEMP_VAL** ME writes the hottest CPU0/CH3 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. ***Note:*** This is byte 9 of EC SMBus Data for PDP Platform PCH Mode Read operation **SDP Platform: DIMM3 TEMP_VAL** ME writes the DIMM3 Data here after collecting it over SMBus. This is in absolute degrees C. ***Note:*** This is byte 9 of EC SMBus Data for SDP Platform PCH Mode Read operation			00h	RO

Table 17-91. Offset B0h: DIMM0—CPU0 DIMM Data (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6		**Offset Start:** B0h **Offset End:** B3h	
Size: 32 bit	**Default:** 00000000h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16		**PDP Platform: CPU0/CH2_DIMM_TEMP_VAL** ME writes the hottest CPU0/CH2 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. ***Note:*** This is byte 8 of EC SMBus Data for PDP Platform PCH Mode Read operation **SDP Platform: DIMM2 TEMP_VAL** ME writes the DIMM2 Data here after collecting it over SMBus. This is in absolute degrees C. ***Note:*** This is byte 8 of EC SMBus Data for SDP Platform PCH Mode Read operation			00h	RO
15 :08		**PDP Platform: CPU0/CH1_DIMM_TEMP_VAL** ME writes the hottest CPU0/CH1 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. ***Note:*** This is byte 7 of EC SMBus Data for PDP Platform PCH Mode Read operation **SDP Platform: DIMM1 TEMP_VAL** ME writes the DIMM1 Data here after collecting it over SMBus. This is in absolute degrees C. ***Note:*** This is byte 7 of EC SMBus Data for SDP Platform PCH Mode Read operation			00h	RO
07 :00		**PDP Platform: CPU0/CH0_DIMM_TEMP_VAL** ME writes the hottest CPU0/CH0 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. ***Note:*** This is byte 6 of EC SMBus Data for PDP Platform PCH Mode Read operation **SDP Platform: DIMM0 TEMP_VAL** ME writes the DIMM0 Data here after collecting it over SMBus. This is in absolute degrees C. ***Note:*** This is byte 6 of EC SMBus Data for SDP Platform PCH Mode Read operation			00h	RO

17.2.3.46 Offset B4h: DIMM1—CPU1 DIMM Data

This register is used by ME FW to store the temperature values of DIMMs connected to CPU1.

Table 17-92. Offset B4h: DIMM1—CPU1 DIMM Data

Description:						
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** B4h **Offset End:** B7h	
Size: 32 bit	**Default:**	00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24		**CPU1/CH3_DIMM_TEMP_VAL** ME writes the hottest for CPU1/CH3 Data here after collecting it over the PECI Bus. This is in absolute degrees C. This is byte 25 of EC SMBus Data for PDP Platform PCH Mode Read operation			00h	RO
23 :16		**CPU1/CH2_DIMM_TEMP_VAL** ME writes the hottest CPU1/CH2 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. This is byte 24 of EC SMBus Data for PDP Platform PCH Mode Read operation			00h	RO
15 :08		**CPU1/CH1_DIMM_TEMP_VAL** ME writes the hottest CPU1/CH1 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. This is byte 23 of EC SMBus Data for PDP Platform PCH Mode Read operation			00h	RO
07 :00		**CPU1/CH0_DIMM_TEMP_VAL** ME writes the hottest CPU1/CH0 DIMM Data here after collecting it over the PECI Bus. This is in absolute degrees C. This is byte 22 of EC SMBus Data for PDP Platform PCH Mode Read operation			00h	RO

17.2.3.47 Offset B8h: DIMMID—DIMM ID

This register is used by ME FW to indicate the hottest DIMMs that have their temperature values stored in DIMM0 (Section 17.2.3.45) and DIMM1 (Section 17.2.3.46) Data registers for CPU0 and CPU1. This register is used in PDP platform applications only.

Table 17-93. Offset B8h: DIMMID—DIMM ID (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16		Reserved		0000h	RO
15 14		**CPU1/CH3_DIMM_ID** DIMM ID for the hottest DIMM for CPU1/CH3. This is the ID for the corresponding DIMM in byte 25 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU1/CH3/DIMM0 01 = CPU1/CH3/DIMM1 10 = CPU1/CH3/DIMM2 11 = No DIMM populated in CPU1/CH3 or DIMMs have no TS.		0h	RO
13 :12		**CPU1/CH2_DIMM_ID** DIMM ID for the hottest DIMM for CPU1/CH2. This is the ID for the corresponding DIMM in byte 24 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU1/CH2/DIMM0 01 = CPU1/CH2/DIMM1 10 = CPU1/CH2/DIMM2 11 = No DIMM populated in CPU1/CH2 or DIMMs have no TS.		0h	RO
11 :10		**CPU1/CH1_DIMM_ID** DIMM ID for the hottest DIMM for CPU1/CH1. This is the ID for the corresponding DIMM in byte 23 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU1/CH1/DIMM0 01 = CPU1/CH1/DIMM1 10 = CPU1/CH1/DIMM2 11 = No DIMM populated in CPU1/CH1 or DIMMs have no TS.		0h	RO
09 :08		**CPU1/CH0_DIMM_ID** DIMM ID for the hottest DIMM for CPU1/CH0. This is the ID for the corresponding DIMM in byte 22 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU1/CH0/DIMM0 01 = CPU1/CH0/DIMM1 10 = CPU1/CH0/DIMM2 11 = No DIMM populated in CPU1/CH0 or DIMMs have no TS.		0h	RO

Table 17-93. Offset B8h: DIMMID—DIMM ID (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06		**CPU0/CH3_DIMM_ID** DIMM ID for the hottest DIMM for CPU0/CH3. This is the ID for the corresponding DIMM in byte 9 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU0/CH3/DIMM0 01 = CPU0/CH3/DIMM1 10 = CPU0/CH3/DIMM2 11 = No DIMM populated in CPU0/CH3 or DIMMs have no TS.		0h	RO
05 :04		**CPU0/CH2_DIMM_ID** DIMM ID for the hottest DIMM for CPU0/CH2. This is the ID for the corresponding DIMM in byte 8 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU0/CH2/DIMM0 01 = CPU0/CH2/DIMM1 10 = CPU0/CH2/DIMM2 11 = No DIMM populated in CPU0/CH2 or DIMMs have no TS.		0h	RO
03 :02		**CPU0/CH1_DIMM_ID** DIMM ID for the hottest DIMM for CPU0/CH1. This is the ID for the corresponding DIMM in byte 7 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU0/CH1/DIMM0 01 = CPU0/CH1/DIMM1 10 = CPU0/CH1/DIMM2 11 = No DIMM populated in CPU0/CH1 or DIMMs have no TS.		0h	RO
01 :00		**CPU0/CH0_DIMM_ID** DIMM ID for the hottest DIMM for CPU0/CH0. This is the ID for the corresponding DIMM in byte 6 of EC SMBus Data for PDP Platform PCH Mode Read operation 00 = CPU0/CH0/DIMM0 01 = CPU0/CH0/DIMM1 10 = CPU0/CH0/DIMM2 11 = No DIMM populated in CPU0/CH0 or DIMMs have no TS.		0h	RO

17.2.3.48 Offset BCh: ECPCLAMP—EC Power Clamp Data

Table 17-94. Offset BCh: ECPCLAMP—EC Power Clamp Data

Description:					
View: PCI	**BAR:** TBARB		**Bus:Device:Function:** B:31:6	**Offset Start:** BCh **Offset End:** BFh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved		0000h	RW
15 :00		MPC CPU Power Clamp Data from EC: — This contains the 16 bits that were received from the EC for the CPU power clamp transaction. This data was used to send a Clamp Power PECI transaction to the cpu.		0000h	RW

17.2.3.49 Offset C2h: TS1PIEN—Thermal Sensor 1 PCI Interrupt Event Enables

This register controls the conditions that result in the PCI Interrupt signal from the Thermal Sensor (TS) logic to assert.

Note: There is a separate enable register per sensor.

Table 17-95. Offset C2h: TS1PIEN—Thermal Sensor 1PCI Interrupt Event Enables

Description:					
View: PCI	**BAR:**	TBARB	**Bus:Device:Function:** B:31:6	**Offset Start:** C2h **Offset End:** C2h	
Size: 8 bit	**Default:**	00h		**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07		Aux2 High-to-Low Enable — See the description for bit 6.		0b	RW
06		Catastrophic High-to-Low Enable — When set to 1, the thermal sensor logic asserts the Thermal logic PCI INTx signal when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in PCI INTx.		0b	RW
05		Hot High-to-Low Enable — See the description for bit 6.		0b	RW
04		Aux High-to-Low Enable — See the description for bit 6.		0b	RW
03		Aux2 Low-to-High Event — See the description for bit 6.		0b	RW
02		Catastrophic Low-to-High Enable — See the description for bit 6.		0b	RW
01		Hot Low-to-High Event — See the description for bit 6.		0b	RW
00		Aux Low-to-High Event — See the description for bit 6.		0b	RW

17.2.3.50 Offset C3h: TS1LOCK—Thermal Sensor 1 Register Lock Controls

This register permits BIOS to lock the contents of the Catastrophic Trip controls and Hot Trip controls. The Catastrophic Trip controls are locked with TSCO bit 7. Also, bit 0 contains the TBCNTEN bit that controls Thermal Burden Counter enable/disable.

Table 17-96. Offset C3h: TS1LOCK—Thermal Sensor 1 Register Lock Controls

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** C3h **Offset End:** C3h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :03	Reserved	Reserved		0h	RW
02	LBH	Hot Trip Lock — This bit, when written to a 1, locks the Hot Trip programming interface, which is bits 15:8 of TSTTP. This bit may only be set to a 0 by a Host Partitioned Reset (bb_rst) reset. Writing a 0 to this bit has no effect. *CF9 warm reset is a Host Partitioned Reset.*		0b	RWL
01 :00	Reserved	Reserved		00b	RW

17.2.3.51 Offset D8h: ITV—Internal Temperature Values

Table 17-97. Offset D8h: ITV—Internal Temperature Values

Description:					
View: PCI	**BAR:** TBARB	**Bus:Device:Function:** B:31:6		**Offset Start:** D8h **Offset End:** DBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved		00h	RW
23 :16		Sequence Number — Provides a sequence number which can be used by the host to detect if ME has hung. The value will roll over to 0 from 0xF. The count is updated at the MPC rate, which is expected to be 200 msec. Host SW can check this value and if it isn't incremented over a second or so, should assume that the ME MCP code is hung. NOTE: if ME is reset, then this value will not change during the reset. After the reset is done, which may take up to 30 seconds, ME may be on line again and this value will start incrementing; indicating that the thermal values are valid again. This is the value of EC byte 10		00h	RO
15 :08	Reserved	Reserved		00h	RW
07 :00		PCH Temperature — ME writes the PCH temperature here. This is in absolute degrees C. This is byte 2 of the EC data.		00h	RO

§ §

18.0 Intel® Management Engine Interface (MEI - B0:D22:F0/F1)

18.1 First Intel Management Engine Interface (MEI1) Configuration Registers (MEI1 — B0:D22:F0)

Table 18-1. MEI1 Configuration Registers Address Map (MEI1—B0:D22:F0) (Sheet 1 of 2)

Offset	Mnemonic & Register Name	Default	Type
00h–01h	"VID—Vendor Identification Register"	8086h	RO
02h–03h	"DID—Device Identification Register"	See register description	RO
04h–05h	"PCICMD—PCI Command Register"	0000h	R/W, RO
06h–07h	"PCISTS—PCI Status Register"	0010h	RO
08h	"RID—Revision Identification Register"	See register description	RO
09h–0Bh	"CC—Class Code Register"	0C8000h	RO
0Eh	"HTYPE—Header Type Register"	00h	RO
10h–17h	"MEI1_MBAR—MEI1 MMIO Base Address Register"	00000000 00000004h	R/W, RO
2Ch–2Dh	"SVID—Subsystem Vendor ID Register"	0000h	R/WO
2Eh–2Fh	"SID—Subsystem ID Register"	0000h	R/WO
34h	"CAPP—Capabilities List Pointer Register"	50h	RO
3Ch–3Dh	"INTR—Interrupt Information Register"	0000h	R/W, RO
3Eh-3Fh	"MLMG—Maximum Latency/Minimum Grant Register"	0000h	RO
40h-43h	"HFS—Host Firmware Status Register"	00000000h	RO
44h-47h	"ME_UMA—Management Engine UMA Register"	00000000h	RO
48-4Bh	"GMES—General ME Status"	00000000h	RO
4Ch-4Fh	"H_GS—Host General Status"	00000000h	RO
50h-51h	"PID—PCI Power Management Capability ID Register"	6001h	RO
52h-53h	"PC—PCI Power Management Capabilities Register"	C803h	RO
54h-55h	"PMCS—PCI Power Management Control and Status Register"	0008h	R/WC, R/W, RO
8Ch-8Dh	"MID—Message Signaled Interrupt Identifiers Register"	0005h	RO
8Eh-8Fh	"MC—Message Signaled Interrupt Message Control Register"	0080h	R/W, RO
90h-93h	"MA—Message Signaled Interrupt Message Address Register"	00000000h	R/W, RO
94h-97h	"MUA—Message Signaled Interrupt Upper Address Register"	00000000h	R/W
98h-99h	"MD—Message Signaled Interrupt Message Data Register"	0000h	R/W

Table 18-1. MEI1 Configuration Registers Address Map (MEI1—B0:D22:F0) (Sheet 2 of 2)

Offset	Mnemonic & Register Name	Default	Type
A0h	"HIDM—MEI Interrupt Delivery Mode"	00h	R/W
BCh-BFh	"HERES—MEI Extend Register Status"	40000000h	RO
C0h-DFh	"HERX—MEI Extend Register DWX"	00000000h	RO

18.1.1 VID—Vendor Identification Register

Address Offset: 00h–01h
Default Value: 8086h
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Vendor ID (VID)** — RO. This is a 16-bit value assigned to Intel.

18.1.2 DID—Device Identification Register

Address Offset: 02h–03h
Default Value: See bit description
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Device ID (DID)** — RO. This is a 16-bit value assigned to the Intel Management Engine Interface controller.

18.1.3 PCICMD—PCI Command Register

Address Offset: 04h–05h
Default Value: 0000h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:11	Reserved
10	**Interrupt Disable (ID)** — R/W. Disables this device from generating PCI line based interrupts. This bit does not have any effect on MSI operation.
9:3	Reserved
2	**Bus Master Enable (BME)**— R/W.**:** Controls the Intel MEI host controller's ability to act as a system memory master for data transfers. When this bit is cleared, Intel ME bus master activity stops and any active DMA engines return to an idle condition. This bit is made visible to firmware through the H_PCI_CSR register, and changes to this bit may be configured by the H_PCI_CSR register to generate an ME MSI. When this bit is 0, Intel MEI is blocked from generating MSI to the host CPU. ***Note:*** This bit does not block Intel MEI accesses to ME-UMA, for example writes or reads to the host and ME circular buffers through the read window and write window registers still cause ME backbone transactions to ME-UMA.
1	**Memory Space Enable (MSE)** — R/W. Controls access to the Intel ME's memory mapped register space. 0 = Disable. Memory cycles within the range specified by the memory base and limit registers are master aborted. 1 = Enable. Allows memory cycles within the range specified by the memory base and limit registers accepted.
0	Reserved

18.1.4 PCISTS—PCI Status Register

Address Offset: 06h–07h
Default Value: 0010h
Attribute: RO
Size: 16 bits

Bit	Description
15:5	Reserved
4	Capabilities List (CL) — RO. Indicates the presence of a capabilities list, hardwired to 1.
3	**Interrupt Status (IS)** — RO. Indicates the interrupt status of the device. 0 = Interrupt is de-asserted. 1 = Interrupt is asserted.
2:0	Reserved

18.1.5 RID—Revision Identification Register

Offset Address: 08h
Default Value: See bit description
Attribute: RO
Size: 8 bits

Bit	Description
7:0	**Revision ID** — RO. 8-bit value indicating the stepping of the SATA* Controller hardware

18.1.6 CC—Class Code Register

Address Offset: 09h–0Bh
Default Value: 078000h
Attribute: RO
Size: 24 bits

Bit	Description
23:16	**Base Class Code (BCC)** — RO. Indicates the base class code of the Intel MEI device.
15:8	**Sub Class Code (SCC)** — RO. Indicates the sub class code of the Intel MEI device.
7:0	**Programming Interface (PI)** — RO. Indicates the programming interface of the Intel MEI device.

18.1.7 HTYPE—Header Type Register

Address Offset: 0Eh
Default Value: 80h
Attribute: RO
Size: 8 bits

Bit	Description
7	**Multi-Function Device (MFD)** — RO. Indicates the Intel MEI host controller is part of a multifunction device.
6:0	Header Layout (HL) — RO. Indicates that the Intel MEI uses a target device layout.

18.1.8 MEI1_MBAR—MEI1 MMIO Base Address Register

Address Offset: 10h–17h
Default Value: 0000000000000004h
Attribute: R/W, RO
Size: 64 bits

This register allocates space for the MEI1 memory mapped registers.

Bit	Description
63:4	**Base Address (BA) — R/W**. Software programs this field with the base address of this region.
3	**Prefetchable Memory (PM)** — RO. Indicates that this range is not pre-fetchable.
2:1	**Type (TP)** — RO. Set to 10b to indicate that this range can be mapped anywhere in 64-bit address space.
0	**Resource Type Indicator (RTE)** — RO. Indicates a request for register memory space.

18.1.9 SVID—Subsystem Vendor ID Register

Address Offset: 2Ch–2Dh
Default Value: 0000h
Attribute: R/WO
Size: 16 bits

Bit	Description
15:0	**Subsystem Vendor ID (SSVID)** — R/WO. Indicates the sub-system vendor identifier. This field should be programmed by BIOS during boot-up. Once written, this register becomes Read Only. This field can only be cleared by PLTRST#.

18.1.10 SID—Subsystem ID Register

Address Offset: 2Eh–2Fh
Default Value: 0000h
Attribute: R/WO
Size: 16 bits

Bit	Description
15:0	**Subsystem ID (SSID)** — R/WO. Indicates the sub-system identifier. This field should be programmed by BIOS during boot-up. Once written, this register becomes Read Only. This field can only be cleared by PLTRST#.

18.1.11 CAPP—Capabilities List Pointer Register

Address Offset: 34h
Default Value: 50h
Attribute: RO
Size: 8 bits

Bit	Description
7:0	**Capabilities Pointer (PTR)** — RO. Indicates that the pointer for the first entry in the capabilities list is at 50h in configuration space.

18.1.12 INTR—Interrupt Information Register

Address Offset: 3Ch–3Dh
Default Value: 0100h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:8	**Interrupt Pin (IPIN)** — RO. This indicates the interrupt pin the Intel MEI host controller uses. The value of 01h selects INTA# interrupt pin.
7:0	**Interrupt Line (ILINE)** — R/W. Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

18.1.13 MLMG—Maximum Latency/Minimum Grant Register

Address Offset: 3Eh–3Fh
Default Value: 0000h
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Maximum Latency/Minimum Grant (MLMG)**— RO. Not used. Hardwired to 0000h.

18.1.14 HFS—Host Firmware Status Register

Address Offset: 40h–43h
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host Firmware Status (HFS)** — RO. This register field is used by Firmware to reflect the operating environment to the host.

18.1.15 ME_UMA—Management Engine UMA Register

Address Offset: 44h–47h
Default Value: 80000000h
Attribute: RO
Size: 32 bits

Bit	Description
31	**Reserved**— RO. Hardwired to 1. Can be used by host software to discover that this register is valid.
30:7	**Reserved**
16	**ME UMA Size Valid -** RO. This bit indicates that FW has written to the MUSZ field.
15:6	**Reserved**
5:0	**ME UMA Size (MUSZ) -** RO. This field reflect ME Firmware's desired size of MEUMA memory region. This field is set by ME firmware prior to core power bringup allowing BIOS to initialize memory. 000000b = 0 MB, No memory allocated to MEUMA 000001b = 1 MB 000010b = 2 MB 000100b = 4 MB 001000b = 8 MB 010000b = 16 MB 100000b = 32 MB

18.1.16 GMES—General ME Status

Address Offset: 48h–4Bh
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**General ME Status (ME_GS)**— RO. This field is populated by ME.

18.1.17 H_GS—Host General Status

Address Offset: 4Ch–4Fh
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host General Status(H_GS)**— RO. General Status of Host, this field is not used by Hardware

18.1.18 PID—PCI Power Management Capability ID Register

Address Offset: 50h–51h
Default Value: 6001h
Attribute: RO
Size: 16 bits

Bit	Description
15:8	**Next Capability (NEXT)** — RO. Value of 60h indicates the location of the next pointer.
7:0	**Capability ID (CID)** — RO. Indicates the linked list item is a PCI Power Management Register.

18.1.19 PC—PCI Power Management Capabilities Register

Address Offset: 52h–53h
Default Value: C803h
Attribute: RO
Size: 16 bits

Bit	Description
15:11	**PME_Support (PSUP)** — RO. This five-bit field indicates the power states in which the function may assert PME#. Intel MEI can assert PME# from any D-state except D1 or D2 which are not supported by Intel MEI.
10:9	Reserved
8:6	**Aux_Current (AC)** — RO. Reports the maximum Suspend well current required when in the $D3_{cold}$ state. Value of 00b is reported.
5	**Device Specific Initialization (DSI)** — RO. Indicates whether device-specific initialization is required.
4	Reserved
3	**PME Clock (PMEC)** — RO. Indicates that PCI clock is not required to generate PME#.
2:0	**Version (VS)** — RO. Hardwired to 011b to indicate support for *Revision 1.2 of the PCI Power Management Specification*.

18.1.20 PMCS—PCI Power Management Control and Status Register

Address Offset: 54h–55h
Default Value: 0008h
Attribute: R/WC, R/W, RO
Size: 16 bits

Bit	Description
15	**PME Status (PMES)** — R/WC. Bit is set by ME Firmware. Host software clears bit by writing '1' to bit.
14:9	Reserved
8	**PME Enable (PMEE)** — R/W. This bit is read/write and is under the control of host SW. It does not directly have an effect on PME events. However, this bit is shadowed so ME FW can monitor it. ME FW will not cause the PMES bit to transition to 1 while the PMEE bit is 0, indicating that host SW had disabled PME. This bit is reset when PLTRST# asserted.
7:4	Reserved
3	**No_Soft_Reset (NSR)** — RO. This bit indicates that when the Intel MEI host controller is transitioning from $D3_{hot}$ to D0 due to a power state command, it does not perform an internal reset. Configuration context is preserved.
2	Reserved
1:0	**Power State (PS)** — R/W. This field is used both to determine the current power state of the Intel MEI host controller and to set a new power state. The values are: 00 = D0 state (default) 11 = $D3_{hot}$ state The D1 and D2 states are not supported for the Intel MEI host controller. When in the $D3_{hot}$ state, the Intel ME's configuration space is available, but the register memory spaces are not. Additionally, interrupts are blocked.

18.1.21 MID—Message Signaled Interrupt Identifiers Register

Address Offset: 8Ch-8Dh
Default Value: 0005h
Attribute: RO
Size: 16 bits

Bit	Description
15:8	**Next Pointer (NEXT)** — RO. Value of 00h indicates that this is the last item in the list.
7:0	**Capability ID (CID)** — RO. Capabilities ID indicates MSI.

18.1.22 MC—Message Signaled Interrupt Message Control Register

Address Offset: 8Eh-8Fh
Default Value: 0080h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:8	Reserved.
7	**64 Bit Address Capable** (C64) — RO. Specifies that function is capable of generating 64-bit messages.
6:1	Reserved
0	**MSI Enable (MSIE)** — R/W. If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts.

18.1.23 MA—Message Signaled Interrupt Message Address Register

Address Offset: 90h-93h
Default Value: 00000000h
Attribute: R/W, RO
Size: 32 bits

Bit	Description
31:2	**Address (ADDR)** — R/W. Lower 32 bits of the system specified message address, always DW aligned.
1:0	Reserved.

18.1.24 MUA—Message Signaled Interrupt Upper Address Register

Address Offset: 94h-97h
Default Value: 00000000h
Attribute: R/W
Size: 32 bits

Bit	Description
31:0	**Upper Address (UADDR)** — R/W. Upper 32 bits of the system specified message address, always DW aligned.

18.1.25 MD—Message Signaled Interrupt Message Data Register

Address Offset: 98h-99h
Default Value: 0000h
Attribute: R/W
Size: 16 bits

Bit	Description
15:0	**Data (DATA)** — R/W. This 16-bit field is programmed by system software if MSI is enabled. Its content is driven during the data phase of the MSI memory write transaction.

18.1.26 HIDM—MEI Interrupt Delivery Mode

Address Offset: A0h
Default Value: 00h
Attribute: R/W
Size: 8 bits

Bit	Description
7:2	Reserved.
1	**MEI Interrupt Delivery Mode (HIDM)** — R/W. These bits control what type of interrupt the Intel MEI will send when ARC writes to set the M_IG bit in AUX space. They are interpreted as follows: 00 = Generate Legacy or MSI interrupt 01 = Generate SCI 10 = Generate SMI
0	**Synchronous SMI Occurrence (SSMIO)** — R/WC. This bit is used by firmware to indicate that a synchronous SMI source has been triggered. Host BIOS SMM handler can use this bit as status indication and clear it once processing is completed. A write of 1 from host SW clears this status bit. ***Note:*** It is possible that an async SMI has occurred prior to sync SMI occurrence and when the BIOS enters the SMM handler, it is possible that both bit 0 and bit 1 of this register could be set.

18.1.27 HERES—MEI Extend Register Status

Address Offset: BCh-BFh
Default Value: 00h
Attribute: RO
Size: 32 bits

Bit	Description
31	**Extend Register Valid (ERV).** Set by firmware after all firmware has been loaded. If ERA field is SHA-1, the result of the extend operation is in HER:5-1. If ERA field is SHA-256, the result of the extend operation is in HER:8-1.
30	**Extend Feature Present (EFP).** This bit is hardwired to 1 to allow driver software to easily detect the chipset supports the Extend Register FW measurement feature.
29:4	**Reserved**
3:0	**Extend Register Algorithm (ERA).** This field indicates the hash algorithm used in the FW measurement extend operations. Encodings are: 0h = SHA-1 2h = SHA-256 Other values = Reserved.

18.1.28 HERX—MEI Extend Register DWX

Address Offset: HER1: C0h-C3h
HER2: C4h-C7h
HER3: C8h-CBh
HER4: CCh-CFh
HER5: D0h-D3h
HER6: D4h-D7h
HER7: D8h-DBh
HER8: DCh-DFh
Attribute: RO
Default Value: 00000000h
Size: 32 bits

Bit	Description
31:0	**Extend Register DWX (ERDWX).** Nth DWORD result of the extend operation. ***Note:*** Extend Operation is HER[5:1] if using SHA-1. If using SHA-2 then Extend Operation is HER[8:1]

18.2 MEI1_MBAR: MEI1 MMIO Registers

These MMIO registers are accessible starting at the MEI1 MMIO Base Address (MEI1_MBAR) which gets programmed into B0:D22:F0:Offset 10-17h. These registers are reset by PLTRST# unless otherwise noted.

Table 18-1. MEI1 MMIO Register Address Map

Offset (BAR= MEI1_MBAR)	Mnemonic & Register Name	Default	Type
00–03h	"H_CB_WW—Host Circular Buffer Write Window"	00000000h	RO
04h-07h	"H_CSR—Host Control Status"	02000000h	RO
08h-0Bh	"ME_CB_RW—ME Circular Buffer Read Window"	00000000h	RO
0Ch-0Fh	"ME CSR_HA—ME Control Status Host Access"	02000000h	RO

18.2.1 H_CB_WW—Host Circular Buffer Write Window

Address Offset: MEI1_MBAR + 00h
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host Circular Buffer Write Window Field (H_CB_WWF)**. This bit field is for host to write into its circular buffer. The host's circular buffer is located at the ME subsystem address specified in the Host CB Base Address register. This field is write only, reads will return arbitrary data. Writes to this register will increment the H_CBWP as long as ME_RDY is 1. When ME_RDY is 0, writes to this register have no effect and are not delivered to the H_CB, nor is H_CBWP incriminated.

18.2.2 H_CSR—Host Control Status

Address Offset: MEI1_MBAR + 04h
Default Value: 02000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:24	**Host Circular Buffer Depth (H_CBD)** — RO. This field indicates the maximum number of 32 bit entries available in the host circular buffer (H_CB). Host software uses this field along with the H_CBRP and H_CBWP fields to calculate the number of valid entries in the H_CB to read or # of entries available for write. This field is implemented with a "1-hot" scheme. Only one bit will be set to a "1" at a time. Each bit position represents the value n of a buffer depth of (2^n). For example, when bit# 1 is 1, the buffer depth is 2; when bit#2 is 1, the buffer depth is 4, etc. The allowed buffer depth values are 2, 4, 8, 16, 32, 64 and 128.
23:16	**Host CB Write Pointer (H_CBWP)**. Points to next location in the H_CB for host to write the data. Software uses this field along with H_CBRP and H_CBD fields to calculate the number of valid entries in the H_CB to read or number of entries available for write.
15:8	**Host CB Read Pointer (H_CBRP)**. Points to next location in the H_CB where a valid data is available for embedded controller to read. Software uses this field along with H_CBWR and H_CBD fields to calculate the number of valid entries in the host CB to read or number of entries available for write.
7:5	**Reserved** Must be programmed to zero
4	**Host Reset (H_RST)**. Setting this bit to 1 will initiate a Intel MEI reset sequence to get the circular buffers into a known good state for host and ME communication. When this bit transitions from 0 to 1, hardware will clear the H_RDY and ME_RDY bits.
3	**Host Ready (H_RDY)**. This bit indicates that the host is ready to process messages.
2	**Host Interrupt Generate (H_IG)**. Once message(s) are written into its CB, the host sets this bit to one for the HW to set the ME_IS bit in the ME_CSR and to generate an interrupt message to ME. HW will send the interrupt message to ME only if the ME_IE is enabled. HW then clears this bit to 0.
1	**Host Interrupt Status (H_IS)**. Hardware sets this bit to 1 when ME_IG bit is set to 1. Host clears this bit to 0 by writing a 1 to this bit position. H_IE has no effect on this bit.
0	**Host Interrupt Enable (H_IE)**. Host sets this bit to 1 to enable the host interrupt (INTR# or MSI) to be asserted when H_IS is set to 1.

18.2.3 ME_CB_RW—ME Circular Buffer Read Window

Address Offset: MEI1_MBAR + 08h
Default Value: FFFFFFFFh
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**ME Circular Buffer Read Window Field (ME_CB_RWF)**. This bit field is for host to read from the ME Circular Buffer. The ME's circular buffer is located at the ME subsystem address specified in the ME CB Base Address register. This field is read only, writes have no effect. Reads to this register will increment the ME_CBRP as long as ME_RDY is 1. When ME_RDY is 0, reads to this register have no effect, all 1s are returned, and ME_CBRP is not incremented.

18.2.4 ME CSR_HA—ME Control Status Host Access

Address Offset: MEI1_MBAR + 0Ch
Default Value: 02000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:24	**ME Circular Buffer Depth Host Read Access (ME_CBD_HRA)**. Host read only access to ME_CBD.
23:16	**ME CB Write Pointer Host Read Access (ME_CBWP_HRA)**. Host read only access to ME_CBWP.
15:8	**ME CB Read Pointer Host Read Access (ME_CBRP_HRA)**. Host read only access to ME_CBRP.
7:5	Reserved
4	**ME Reset Host Read Access (ME_RST_HRA)**. Host read access to ME_RST.
3	**ME Ready Host Read Access (ME_RDY_HRA):** Host read access to ME_RDY.
2	**ME Interrupt Generate Host Read Access (ME_IG_HRA)**. Host read only access to ME_IG.
1	**ME Interrupt Status Host Read Access (ME_IS_HRA)**. Host read only access to ME_IS.
0	**ME Interrupt Enable Host Read Access (ME_IE_HRA)**. Host read only access to ME_IE.

18.3 Second Management Engine Interface (MEI2) Configuration Registers (MEI2—B0:D22:F1)

Table 18-2. MEI2 Configuration Registers Address Map (MEI2—B0:D22:F1)

Offset	Mnemonic & Register Name	Default	Type
00h–01h	"VID—Vendor Identification Register"	8086h	RO
02h–03h	"DID—Device Identification Register"	See register description	RO
04h–05h	"PCICMD—PCI Command Register"	0000h	R/W, RO
06h–07h	"PCISTS—PCI Status Register"	0010h	RO
08h	"RID—Revision Identification Register"	See register description	RO
09h–0Bh	"CC—Class Code Register"	0C8000h	RO
0Eh	"HTYPE—Header Type Register"	00h	RO
10h–17h	"MEI_MBAR—MEI MMIO Base Address Register"	00000000 00000004h	R/W, RO
2Ch–2Dh	"SVID—Subsystem Vendor ID Register"	0000h	R/WO
2Eh–2Fh	"SID—Subsystem ID Register"	0000h	R/WO
34h	"CAPP—Capabilities List Pointer Register"	50h	RO
3Ch–3Dh	"INTR—Interrupt Information Register"	0000h	R/W, RO
3Eh–3Fh	"MLMG—Maximum Latency/Minimum Grant Register"	0000h	RO
40h–43h	"HFS—Host Firmware Status Register"	00000000h	RO
48–4Bh	"GMES—General ME Status"	00000000h	RO
4Ch–4Fh	"H_GS—Host General Status"	00000000h	RO
50h–51h	"PID—PCI Power Management Capability ID Register"	6001h	RO
52h–53h	"PC—PCI Power Management Capabilities Register"	C803h	RO
54h–55h	"PMCS—PCI Power Management Control and Status Register"	0008h	R/WC, R/W, RO
8Ch–8Dh	"MID—Message Signaled Interrupt Identifiers Register"	0005h	RO
8Eh–8Fh	"MC—Message Signaled Interrupt Message Control Register"	0080h	R/W, RO
90h–93h	"MA—Message Signaled Interrupt Message Address Register"	00000000h	R/W, RO
94h–97h	"MUA—Message Signaled Interrupt Upper Address Register"	00000000h	R/W
98h–99h	"MD—Message Signaled Interrupt Message Data Register"	0000h	R/W
A0h	"HIDM—MEI Interrupt Delivery Mode"	00h	R/W
BC–BF	"HERES—MEI Extend Register Status"	40000000h	RO
C0–DF	"HERX—MEI Extend Register DWX"	00000000h	RO

18.3.1 VID—Vendor Identification Register

Address Offset: 00h–01h
Default Value: 8086h
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Vendor ID (VID)** — RO. This is a 16-bit value assigned to Intel.

18.3.2 DID—Device Identification Register

Address Offset: 02h–03h
Default Value: See bit description
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Device ID (DID)** — RO. This is a 16-bit value assigned to the Intel Management Engine Interface controller.

18.3.3 PCICMD—PCI Command Register

Address Offset: 04h–05h
Default Value: 0000h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:11	Reserved
10	**Interrupt Disable (ID)** — R/W. Disables this device from generating PCI line based interrupts. This bit does not have any effect on MSI operation.
9:3	Reserved
2	**Bus Master Enable (BME)**— R/W. Controls the Intel MEI host controller's ability to act as a system memory master for data transfers. When this bit is cleared, Intel MEI bus master activity stops and any active DMA engines return to an idle condition. This bit is made visible to firmware through the H_PCI_CSR register, and changes to this bit may be configured by the H_PCI_CSR register to generate an ME MSI. When this bit is 0, Intel MEI is blocked from generating MSI to the host CPU. ***Note:*** This bit does not block Intel MEI accesses to ME-UMA, for example writes or reads to the host and ME circular buffers through the read window and write window registers still cause ME backbone transactions to ME-UMA.
1	**Memory Space Enable (MSE)** — R/W. Controls access to the Intel ME's memory mapped register space. 0 = Disable. Memory cycles within the range specified by the memory base and limit registers are master aborted. 1 = Enable. Allows memory cycles within the range specified by the memory base and limit registers accepted.
0	Reserved

18.3.4 PCISTS—PCI Status Register

Address Offset: 06h–07h
Default Value: 0010h
Attribute: RO
Size: 16 bits

Bit	Description
15:5	Reserved
4	Capabilities List (CL) — RO. Indicates the presence of a capabilities list, hardwired to 1.
3	**Interrupt Status** — RO. Indicates the interrupt status of the device. 0 = Interrupt is de-asserted. 1 = Interrupt is asserted.
2:0	Reserved

18.3.5 RID—Revision Identification Register

Offset Address: 08h
Default Value: See bit description
Attribute: RO
Size: 8 bits

Bit	Description
7:0	**Revision ID** — RO. 8-bit value indicating the stepping of the SATA* Controller hardware.

18.3.6 CC—Class Code Register

Address Offset: 09h–0Bh
Default Value: 078000h
Attribute: RO
Size: 24 bits

Bit	Description
23:16	**Base Class Code (BCC)** — RO. Indicates the base class code of the Intel MEI device.
15:8	**Sub Class Code (SCC)** — RO. Indicates the sub class code of the Intel MEI device.
7:0	**Programming Interface (PI)** — RO. Indicates the programming interface of the Intel MEI device.

18.3.7 HTYPE—Header Type Register

Address Offset: 0Eh
Default Value: 80h
Attribute: RO
Size: 8 bits

Bit	Description
7	**Multi-Function Device** (MFD) — RO. Indicates the Intel MEI host controller is part of a multifunction device.
6:0	Header Layout (HL) — RO. Indicates that the Intel MEI uses a target device layout.

18.3.8 MEI_MBAR—MEI MMIO Base Address Register

Address Offset: 10h–17h
Default Value: 0000000000000004h
Attribute: R/W, RO
Size: 64 bits

This register allocates space for the Intel MEI memory mapped registers.

Bit	Description
63:4	**Base Address (BA) — R/W**. Software programs this field with the base address of this region.
3	**Prefetchable Memory (PM)** — RO. Indicates that this range is not pre-fetchable.
2:1	**Type (TP)** — RO. Set to 10b to indicate that this range can be mapped anywhere in 64-bit address space.
0	**Resource Type Indicator (RTE)** — RO. Indicates a request for register memory space.

18.3.9 SVID—Subsystem Vendor ID Register

Address Offset: 2Ch–2Dh
Default Value: 0000h
Attribute: R/WO
Size: 16 bits

Bit	Description
15:0	**Subsystem Vendor ID (SSVID)** — R/WO. Indicates the sub-system vendor identifier. This field should be programmed by BIOS during boot-up. Once written, this register becomes Read Only. This field can only be cleared by PLTRST#.

18.3.10 SID—Subsystem ID Register

Address Offset: 2Eh–2Fh
Default Value: 0000h
Attribute: R/WO
Size: 16 bits

Bit	Description
15:0	**Subsystem ID (SSID)** — R/WO. Indicates the sub-system identifier. This field should be programmed by BIOS during boot-up. Once written, this register becomes Read Only. This field can only be cleared by PLTRST#.

18.3.11 CAPP—Capabilities List Pointer Register

Address Offset: 34h
Default Value: 50h
Attribute: RO
Size: 8 bits

Bit	Description
7:0	**Capabilities Pointer (PTR)** — RO. Indicates that the pointer for the first entry in the capabilities list is at 50h in configuration space.

18.3.12 INTR—Interrupt Information Register

Address Offset: 3Ch–3Dh
Default Value: 0100h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:8	Interrupt Pin (IPIN) — RO. This field indicates the interrupt pin the Intel MEI host controller uses. The value of 01h selects INTA# interrupt pin.
7:0	**Interrupt Line (ILINE)** — R/W. Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

18.3.13 MLMG—Maximum Latency/Minimum Grant Register

Address Offset: 3Eh–3Fh
Default Value: 0000h
Attribute: RO
Size: 16 bits

Bit	Description
15:0	**Maximum Latency/Minimum Grant (MLMG)**— RO. Not used. Hardwired to 0000h.

18.3.14 HFS—Host Firmware Status Register

Address Offset: 40h–43h
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host Firmware Status (HFS)** — RO. This register field is used by Firmware to reflect the operating environment to the host.

18.3.15 GMES—General ME Status

Address Offset: 48h–4Bh
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**General ME Status(ME_GS)**— RO. This field is populated by ME.

18.3.16 H_GS—Host General Status

Address Offset: 4Ch–4Fh
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host General Status(H_GS)**— RO. General Status of Host, this field is not used by Hardware

18.3.17 PID—PCI Power Management Capability ID Register

Address Offset: 50h–51h
Default Value: 6001h
Attribute: RO
Size: 16 bits

Bit	Description
15:8	**Next Capability (NEXT)** — RO. Value of 60h indicates the location of the next pointer.
7:0	**Capability ID (CID)** — RO. Indicates the linked list item is a PCI Power Management Register.

18.3.18 PC—PCI Power Management Capabilities Register

Address Offset: 52h–53h
Default Value: C803h
Attribute: RO
Size: 16 bits

Bit	Description
15:11	**PME_Support (PSUP)** — RO. This five-bit field indicates the power states in which the function may assert PME#. Intel MEI can assert PME# from any D-state except D1 or D2 which are not supported by Intel MEI.
10:9	Reserved
8:6	**Aux_Current (AC)** — RO. Reports the maximum Suspend well current required when in the $D3_{cold}$ state. Value of 00b is reported.
5	**Device Specific Initialization (DSI)** — RO. Indicates whether device-specific initialization is required.
4	Reserved
3	**PME Clock (PMEC)** — RO. Indicates that PCI clock is not required to generate PME#.
2:0	**Version (VS)** — RO. Hardwired to 011b to indicate support for *Revision 1.2 of the PCI Power Management Specification*.

18.3.19 PMCS—PCI Power Management Control and Status Register

Address Offset: 54h–55h
Default Value: 0008h
Attribute: R/WC, R/W, RO
Size: 16 bits

Bit	Description
15	**PME Status (PMES)** — R/WC. Bit is set by ME Firmware. Host software clears bit by writing 1 to bit.
14:9	Reserved
8	**PME Enable (PMEE)** — R/W. This bit is read/write and is under the control of host SW. It does not directly have an effect on PME events. However, this bit is shadowed so ME FW can monitor it. ME FW will not cause the PMES bit to transition to 1 while the PMEE bit is 0, indicating that host SW had disabled PME. This bit is reset when PLTRST# asserted.
7:4	Reserved
3	**No_Soft_Reset (NSR)** — RO. This bit indicates that when the Intel MEI host controller is transitioning from $D3_{hot}$ to D0 due to a power state command, it does not perform an internal reset. Configuration context is preserved.
2	Reserved
1:0	**Power State (PS)** — R/W. This field is used both to determine the current power state of the Intel MEI host controller and to set a new power state. The values are: 00 = D0 state (default) 11 = $D3_{hot}$ state The D1 and D2 states are not supported for the Intel MEI host controller. When in the $D3_{hot}$ state, the Intel ME's configuration space is available, but the register memory spaces are not. Additionally, interrupts are blocked.

18.3.20 MID—Message Signaled Interrupt Identifiers Register

Address Offset: 8Ch-8Dh
Default Value: 0005h
Attribute: RO
Size: 16 bits

Bit	Description
15:8	**Next Pointer (NEXT)** — RO. Value of 00h indicates that this is the last item in the list.
7:0	**Capability ID (CID)** — RO. Capabilities ID indicates MSI.

18.3.21 MC—Message Signaled Interrupt Message Control Register

Address Offset: 8Eh-8Fh
Default Value: 0080h
Attribute: R/W, RO
Size: 16 bits

Bit	Description
15:8	Reserved.
7	**64 Bit Address Capable** (C64) — RO. Specifies that function is capable of generating 64-bit messages.
6:1	Reserved
0	**MSI Enable (MSIE)** — R/W. If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts.

18.3.22 MA—Message Signaled Interrupt Message Address Register

Address Offset: 90h-93h
Default Value: 00000000h
Attribute: R/W, RO
Size: 32 bits

Bit	Description
31:2	**Address (ADDR)** — R/W. Lower 32 bits of the system specified message address, always DW aligned.
1:0	Reserved.

18.3.23 MUA—Message Signaled Interrupt Upper Address Register

Address Offset: 94h-97h
Default Value: 00000000h
Attribute: R/W
Size: 32 bits

Bit	Description
31:0	**Upper Address (UADDR)** — R/W. Upper 32 bits of the system specified message address, always DW aligned.

18.3.24 MD—Message Signaled Interrupt Message Data Register

Address Offset: 98h-99h
Default Value: 0000h
Attribute: R/W
Size: 16 bits

Bit	Description
15:0	**Data (DATA)** — R/W. This 16-bit field is programmed by system software if MSI is enabled. Its content is driven during the data phase of the MSI memory write transaction.

18.3.25 HIDM—MEI Interrupt Delivery Mode

Address Offset: A0h
Default Value: 00h
Attribute: R/W
Size: 8 bits

Bit	Description
7:2	Reserved.
1	**Intel MEI Interrupt Delivery Mode (HIDM)** — R/W. These bits control what type of interrupt the Intel MEI will send when ARC writes to set the M_IG bit in AUX space. They are interpreted as follows: 00 = Generate Legacy or MSI interrupt 01 = Generate SCI 10 = Generate SMI
0	**Synchronous SMI Occurrence (SSMIO)** — R/WC. This bit is used by firmware to indicate that a synchronous SMI source has been triggered. Host BIOS SMM handler can use this bit as status indication and clear it once processing is completed. A write of 1 from host SW clears this status bit. ***Note:*** It is possible that an async SMI has occurred prior to sync SMI occurrence and when the BIOS enters the SMM handler, it is possible that both bit 0 and bit 1 of this register could be set.

18.3.26 HERES—MEI Extend Register Status

Address Offset: BCh-BFh
Default Value: 00h
Attribute: RO
Size: 32 bits

Bit	Description
31	**Extend Register Valid (ERV).** Set by firmware after all firmware has been loaded. If ERA field is SHA-1, the result of the extend operation is in HER:5-1. If ERA field is SHA-256, the result of the extend operation is in HER:8-1.
30	**Extend Feature Present (EFP).** This bit is hardwired to 1 to allow driver software to easily detect the chipset supports the Extend Register FW measurement feature.
29:4	**Reserved**
3:0	**Extend Register Algorithm (ERA).** This field indicates the hash algorithm used in the FW measurement extend operations. Encodings are: 0h = SHA-1 2h = SHA-256 Other values = Reserved.

18.3.27 HERX—MEI Extend Register DWX

Address Offset: HER1: C0h-C3h
HER2: C4h-C7h
HER3: C8h-CBh
HER4: CCh-CFh
HER5: D0h-D3h
HER6: D4h-D7h
HER7: D8h-DBh
HER8: DCh-DFh
Attribute: RO
Default Value: 00000000h
Size: 32 bits

Bit	Description
31:0	**Extend Register DWX (ERDWX):** Xth DWORD result of the extend operation. ***Note:*** Extend Operation is HER[5:1] if using SHA-1. If using SHA-2, then Extend Operation is HER[8:1]

18.4 MEI2_MBAR—MEI2 MMIO Registers

These MMIO registers are accessible starting at the MEI2 MMIO Base Address (MEI2_MBAR) which gets programmed into B0:D22:F1:Offset 10-17h. These registers are reset by PLTRST# unless otherwise noted.

Table 18-3. MEI2 MMIO Register Address Map

Offset (BAR= MEI2_MBAR)	Mnemonic & Register Name	Default	Type
00–03h	"H_CB_WW—Host Circular Buffer Write Window"	00000000h	RO
04h-07h	"H_CSR—Host Control Status"	02000000h	RO
08h-0Bh	"ME_CB_RW—ME Circular Buffer Read Window"	00000000h	RO
0Ch-0Fh	"ME CSR_HA—ME Control Status Host Access"	02000000h	RO

18.4.1 H_CB_WW—Host Circular Buffer Write Window

Address Offset: MEI2_MBAR + 00h
Default Value: 00000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:0	**Host Circular Buffer Write Window Field (H_CB_WWF)**. This bit field is for host to write into its circular buffer. The host's circular buffer is located at the ME subsystem address specified in the Host CB Base Address register. This field is write only, reads will return arbitrary data. Writes to this register will increment the H_CBWP as long as ME_RDY is 1. When ME_RDY is 0, writes to this register have no effect and are not delivered to the H_CB, nor is H_CBWP incremented.

18.4.2 H_CSR—Host Control Status

Address Offset: MEI2_MBAR + 04h
Default Value: 02000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:24	**Host Circular Buffer Depth (H_CBD) — RO**. This field indicates the maximum number of 32 bit entries available in the host circular buffer (H_CB). Host software uses this field along with the H_CBRP and H_CBWP fields to calculate the number of valid entries in the H_CB to read or # of entries available for write. ***Note:*** This field is implemented with a "1-hot" scheme. Only one bit will be set to a 1 at a time. Each bit position represents the value n of a buffer depth of (2^n). For example, when bit# 1 is 1, the buffer depth is 2; when bit#2 is 1, the buffer depth is 4, etc. The allowed buffer depth values are 2, 4, 8, 16, 32, 64 and 128.
23:16	**Host CB Write Pointer (H_CBWP)**. Points to next location in the H_CB for host to write the data. Software uses this field along with H_CBRP and H_CBD fields to calculate the number of valid entries in the H_CB to read or number of entries available for write.
15:8	**Host CB Read Pointer (H_CBRP)**. Points to next location in the H_CB where a valid data is available for embedded controller to read. Software uses this field along with H_CBWR and H_CBD fields to calculate the number of valid entries in the host CB to read or number of entries available for write.
7:5	**Reserved** Must be programmed to zero
4	**Host Reset (H_RST)**. Setting this bit to 1 will initiate a Intel MEI reset sequence to get the circular buffers into a known good state for host and ME communication. When this bit transitions from 0 to 1, hardware will clear the H_RDY and ME_RDY bits.
3	**Host Ready (H_RDY)**. This bit indicates that the host is ready to process messages.
2	**Host Interrupt Generate (H_IG)**. Once message(s) are written into its CB, the host sets this bit to one for the HW to set the ME_IS bit in the ME_CSR and to generate an interrupt message to ME. HW will send the interrupt message to ME only if the ME_IE is enabled. HW then clears this bit to 0.
1	**Host Interrupt Status (H_IS)**. Hardware sets this bit to 1 when ME_IG bit is set to 1. Host clears this bit to 0 by writing a 1 to this bit position. H_IE has no effect on this bit.
0	**Host Interrupt Enable (H_IE)**. Host sets this bit to 1 to enable the host interrupt (INTR# or MSI) to be asserted when H_IS is set to 1.

18.4.3 ME_CB_RW—ME Circular Buffer Read Window

Address Offset: MEI2_MBAR + 08h
Default Value: FFFFFFFFh
Attribute: RO
Size: 32 bits

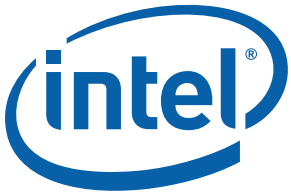
Bit	Description
31:0	**ME Circular Buffer Read Window Field (ME_CB_RWF)**. This bit field is for host to read from the ME Circular Buffer. The ME's circular buffer is located at the ME subsystem address specified in the ME CB Base Address register. This field is read only, writes have no effect. Reads to this register will increment the ME_CBRP as long as ME_RDY is 1. When ME_RDY is 0, reads to this register have no effect, all 1s are returned, and ME_CBRP is not incremented.

18.4.4 ME CSR_HA—ME Control Status Host Access

Address Offset: MEI2_MBAR + 0Ch
Default Value: 02000000h
Attribute: RO
Size: 32 bits

Bit	Description
31:24	**ME Circular Buffer Depth Host Read Access (ME_CBD_HRA)**. Host read only access to ME_CBD.
23:16	**ME CB Write Pointer Host Read Access (ME_CBWP_HRA)**. Host read only access to ME_CBWP.
15:8	**ME CB Read Pointer Host Read Access (ME_CBRP_HRA)**. Host read only access to ME_CBRP.
7:5	Reserved
4	**ME Reset Host Read Access (ME_RST_HRA)**. Host read access to ME_RST.
3	**ME Ready Host Read Access (ME_RDY_HRA)**. Host read access to ME_RDY.
2	**ME Interrupt Generate Host Read Access (ME_IG_HRA)**. Host read only access to ME_IG.
1	**ME Interrupt Status Host Read Access (ME_IS_HRA)**. Host read only access to ME_IS.
0	**ME Interrupt Enable Host Read Access (ME_IE_HRA)**. Host read only access to ME_IE.

§ §



PCIe* EndPoint & GbE-Volume 2 of 4

April 2014

Order Number: 327879-005US

§ §

19.0 PCI Express* EndPoint Introduction

19.1 PCI Express* EndPoint (EP)

19.1.1 Overview

The PCIe* EP provides an interface for the system to connect to a PCIe* Root Complex. It implements the EP functionality, including the Type 0 configuration and the PCIe* extended configuration that enable a host CPU to enumerate, configure and use Intel® QuickAssist Technology.

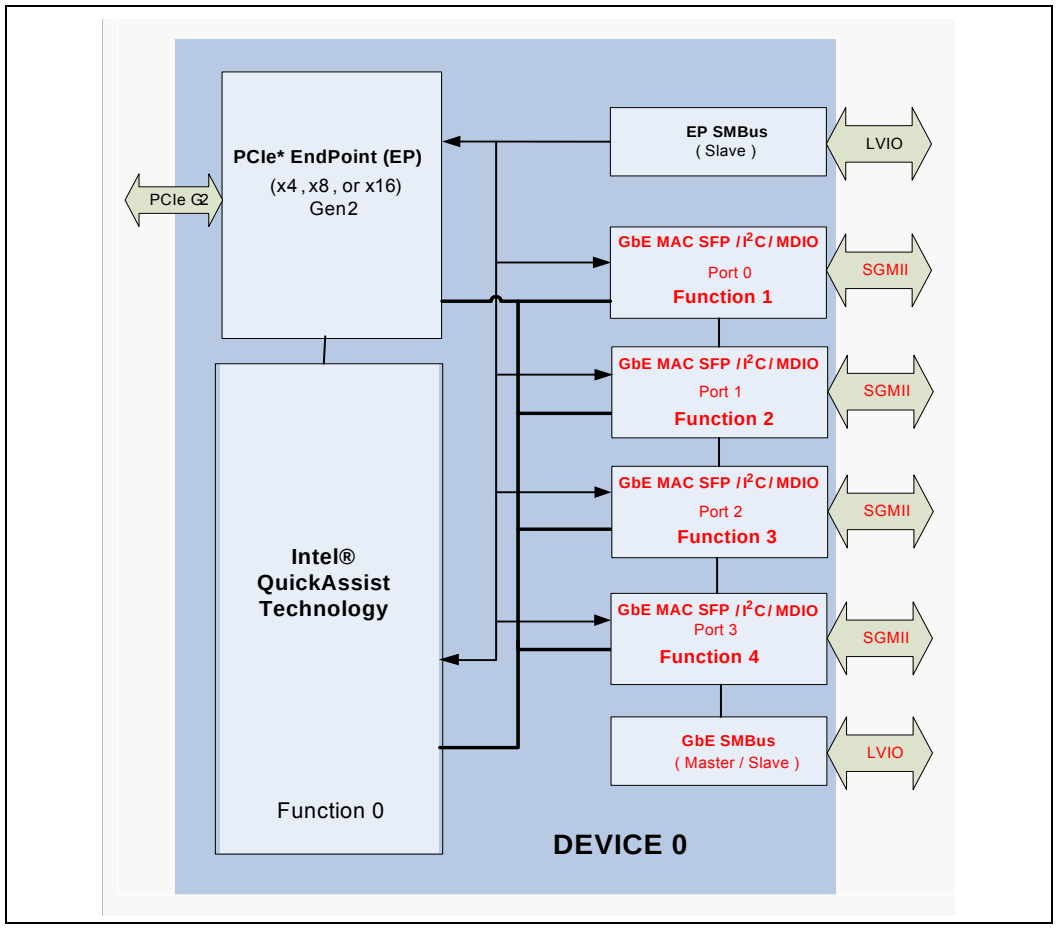
19.1.2 Feature List

- Type 0 configuration and extended PCIe* configuration registers
  - — Materializes as a Multi-function PCIe* EndPoint
  - — Implements Type 0 PCIe* configuration space registers
  - — Interfaces to Gen 2 PCIe* and is compliant to PCIe* 2.0 Base Specification
  - — Generates INT[A,B,C,D] messages Collates interrupts from different sources in the EP itself and other agents and generates INTx messages, MSI, or MSI-X.
- Single Root I/O Virtualization (SR-IOV for Intel® QuickAssist Technology only) Extended Capability Record
  - — 1 Physical Function (INTx, MSI, MSI-X)
  - — 16 Virtual Functions (MSI only)

19.1.3 EP Interface Block Diagram

Figure 19-1 shows the block diagram of the EP Interface.

Figure 19-1. PCIe* EP Interface Block Diagram

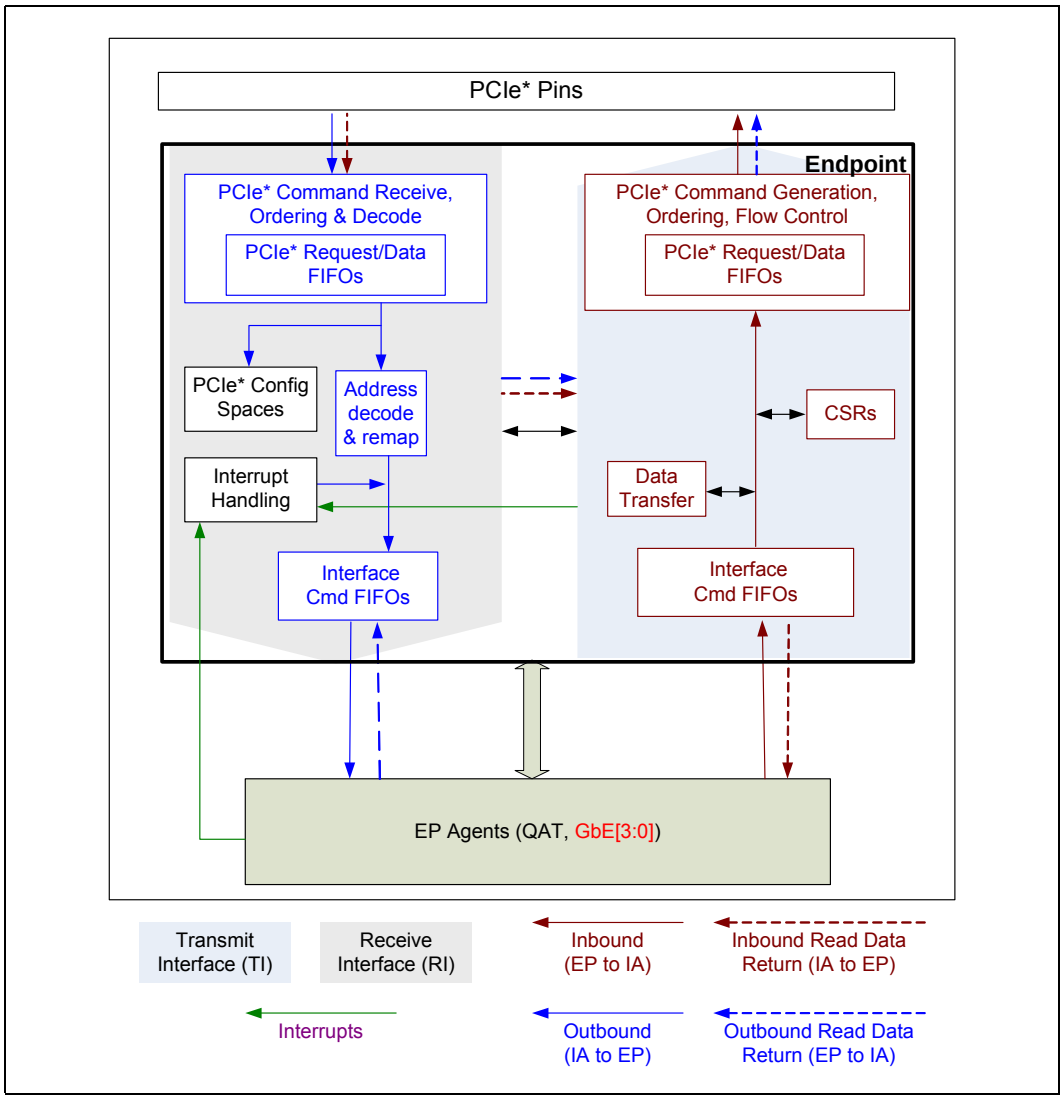


Note: Functions in **RED** are **NOT** available in the DH89xxCL devices.

19.1.4 EP Functional Description

Figure 19-2 shows the functional description block diagram of the EP Interface.

Figure 19-2. EP Functional Description Block Diagram



Note: Functions in **RED** are **NOT** available in the DH89xxCL devices.

19.1.4.1 Transmit Interface (TI)

TI is responsible for converting EP Agents' commands towards system memory and OB completions into the PCIe* transactions. TI connects to the inbound port of the PCIe* root complex.

19.1.4.2 Receive Interface (RI)

The Receive Interface (RI) is responsible for converting OB PCIe* commands from the IA cores into EP Agents' commands. OB read completions for PCIe* commands initiated by the TI are also received by the RI.

19.1.4.3 PCIe* EP Functions

RI connects to the OB port of the PCIe* root complex. Table 19-1 lists the RI supported PCIe* transactions. The EP only supports requests that are aligned on DWORD address boundaries.

Table 19-1. PCIe* Commands Supported by RI

PCIe* Command	Description	Source	EP Attributes	Comments
MRd	Memory Read	IA reading EP MMIO region, PCIe* extended Configuration space	4B	
MWr	Memory Write	IA writing to EP MMIO region, PCIe* extended configuration space	4-8B	
IORd	IO Read	IA reading GbE[1] IO space	4B	
IOWr	IO Write	IA writing GbE[1] IO space	4B	Note 1
CfgRd0	Type 0 Configuration Read	IA reading EP Configuration Register	1-4B	
CfgWr0	Type 0 Configuration Write	Non-Posted IA writing to EP Configuration Register.	1-4B	Note 2
CmpID	Completion with Data	Read Completion with Data for a Read Command initiated by TI	4B-256B	
Msg	Messages without Data	Root Complex	Power Management Messages	

Note:
1. IO transactions are supported for GbEs to access the IOADDR/IODATA registers.
2. This is a non-posted transaction. RI will generate a completion without data after the configuration write is completed. The completion will be routed via a RI-TI private bus to the TI. The TI will initiate the PCIe* completion packet.

1. This GbE interface is not available in the DH89xxCL devices.

19.1.4.4 EP Function Mapping

The EP implements a multi-function PCIe* device. Table 19-2 lists the mapping of the EP functionality.

Table 19-2. PCIe* EP Function Mapping

EP Cluster	Functional Blocks	PCIe* Function Number	BARs	Interrupt Capability	PCIe* Extended/PCI Capabilities
Intel® QuickAssist Technology	QAT	0	PeQATBAR, PMISCBAR, PETRINGCSRBAR[1]	MSI, MSI-X with 17 vectors, INTA	MSI, MSI-X, PM, PCIe*, AER,
GbE[0][2]	GbE	1	GBEPCIBAR0[0:3] GBEPCIBAR1[0:3] GBEPCIBAR2[0:3][3]	MSI, MSI-X with 10 vectors, INTB	MSI, MSI-X, PM, PCIe*, AER
GbE[1][2]		2		MSI, MSI-X with 10 vectors, INTC	
GbE[2][2]		3		MSI, MSI-X with 10 vectors, INTD	
GbE[3][2]		4		MSI, MSI-X with 10 vectors, INTB	

1. PMISCBAR is used for accessing EP, QAT, MSI-X Table, etc.
2. This GbE interface is not available in the DH89xxCL devices.
3. Based on GbE. BARs points to CSRs and MSI-X Tables.

19.1.4.5 EP Mapping of BARs to MMIO

Table 19-3 shows the mapping of the EP MMIO regions to the BARs. See each PCIe* function BAR definition for more details.

Table 19-3. EP Mapping of BARs to MMIOs

Function #	BAR Name (Size)	Region Size	Functionality	Comments
0	PeQATBAR, PMISCBAR, PETRINGCSRBAR (128KB)	120KB	EP, QAT	This region is mapped to the EP and QAT CSRs
0		4KB	EP MMIO CSRs	The EP CSRs are located in the EP.
0		4KB	MSI-X Tables	The MSI-X Tables are located in the EP.
1, 2, 3, 4 (4 GbEs)[1]	GBEPCIBAR0[0:3]	128KB	GbE internal CSRs	See Chapter 29.0, "Function 1-4 (GbE)". This BAR is used to access the GbE Internal CSRs.
	GBEPCIBAR1[0:3]	32B	GbE internal CSRs.	See Chapter 29.0, "Function 1-4 (GbE)". This IO BAR is used to access the GbE internal CSRs.
	GBEPCIBAR2[0:3]	16KB	MSI-X Tables	The MSI-X Tables are located in the EP. See Chapter 29.0, "Function 1-4 (GbE)".
Note: Accesses to MMIO region that are not implemented within the regions defined in this table will return 0's. Accesses to MMIO regions that is not claimed by any of the BARs will return all 1s.				

1. These GbE interfaces are not available in the DH89xxCL devices.

19.1.5 PCIe* EP Interrupts

The RI is responsible for collating interrupts and errors from the EndPoint and taking the appropriate actions based on the PCIe* configurations status.

Interrupts from EP can be configured in the RI to be routed to the IA cores. The interrupts can be sent upstream as either an MSI, MSI-X, or by using INTx virtual wire interrupt signaling mechanism (which are messages on PCIe*).

19.1.5.1 INTx

PCIe* supports devices that need to use the PCI compatible legacy interrupt signaling: INT[A,B,C,D]. Instead of dedicated pins for this functionality, PCIe* supports inband virtual wires.

The EP supports generation of INTx virtual wire interrupt signalling based on internal generated interrupts. The INTx notification is accomplished using PCIe* messages.

The RI is responsible for collating interrupt sources from the EP Agents and generating the transaction that is translated to a PCIe* interrupt message to the IA. Upon detecting an interrupt from an internal source, the RI first determines whether to generate an MSI, MSI-X, or INTx message.

19.1.5.2 MSI-X

MSI-X defines a separate optional extension to the basic MSI functionality. Compared to MSI, MSI-X supports a larger number of vectors per function, the ability for the OS to program independent address and data values for each vector. The address and data values for each vector is specified by a table that resides in EP MMIO space.

Once an MSI has been generated using MSI-X, its characteristics are similar to an MSI that was generated using an MSI capability record.

MSI-X however, supports per-vector masking that is optional for MSI. MSI-X also supports a Function Mask bit, which when set masks all of the vectors associated with a function. Per-vector masking is managed through a Mask and Pending bit pair per MSI vector or MSI-X Table entry. An MSI vector is masked when its associated Mask bit is set. An MSI-X vector is masked when its associated MSI-X Table entry Mask bit or the MSI-X Function Mask bit is set. While a vector is masked, the function is prohibited from sending the associated message, and the function must set the associated Pending bit whenever the function would otherwise send the message. When software unmasks a vector whose associated Pending bit is set, the function must schedule sending the associated message, and clear the Pending bit as soon as the message has been sent.

When MSI-X is enabled, EP generates an interrupt using a PCIe* posted memory write transaction. The address and data of that transaction are determined by the system and programmed in PCI MSI-X table entry.

19.1.6 PCIe* EP Errors

There are two classes of errors that the EP will encounter:

- Errors that occur due to PCIe* transactions
  - PCIe* error management defines the scope of these kinds of errors including how they are reported and handled.
- Other errors not directly related to PCIe* transactions
  - These class of errors are reported via interrupts and do not fall under the scope of PCIe* error management mechanisms

19.1.6.1 PCIe* Error Management

PCIe* error management focuses on errors associated with the PCIe* interface and the transactions between the transaction layers of the transmitting and the receiving PCIe* ports.

19.1.6.2 PCIe* Error Reporting Mechanisms

The PCIe* specification provides three mechanisms for reporting errors that occur while servicing to PCIe* transactions.

1. PCI
   a. This mechanism provides backward compatibility with legacy PCI compatible software and is required per the PCIe* specification. Legacy compatible PCI software will use this mechanism and control this feature via the PCI configuration Command Register. When this mechanism is enabled, EP will also log error status information in the PCI Configuration Status register.
2. PCIe*
   a. This is the baseline PCIe* mechanism for software that understands PCIe* devices. The mechanism can be enabled via the PCIe* Device Control Register. Error status is logged in the PCIe* Device status register.
   b. Supported error classifications: Unsupported request type, fatal error, non-fatal error, and correctable error.
3. PCIe* AER
   a. The optional advanced error reporting registers can be implemented by PCIe* devices. This mechanism is defined using a PCIe* extended capability structure.

b. All EP Functions will support AER.

19.1.6.3 PCIe* Error Handling and Signalling

The EP will classify errors in the following three PCIe* specification-defined buckets:

1. Correctable errors: These errors are handled by the HW.
2. Uncorrectable errors (non-fatal): Handled by device-specific software
3. Uncorrectable errors (fatal): Handled by system software

EP will use three different mechanisms to signal errors that occur when processing PCIe* commands.

1. Completion status
2. Error forwarding or data poisoning
3. Error messages

19.1.6.4 PCIe* Error Sources

Table 19-4 defines the standard PCIe* defined error sources. The following sections describes each of the PCIe* error types, and describes how the EP handles the PCIe* errors.

Table 19-4. PCIe* Error Sources

	Type of Error	Reported Using	Comment
TL	ECRC Check		Not supported in EP
	Malformed TLP	Uncorrectable Fatal Message	RI will perform checks on PCIe* transactions received from IA.
	Completion Timeout	Uncorrectable Non-Fatal Message	RI will generate these Split Transaction Errors. Completer Abort is supported in RI.
	Unsupported Requests	Completion Status	
	Completer Abort (optional)		
	Unexpected Completion	Uncorrectable Non-Fatal Message	
	Data Corruption/Poisoning	PCIe* Header	TI will set the EP bit in the header based on data error. Writes that terminate in RI get dropped and status logged. Writes that target a EP function will be sent with data error and logged in RI.
	Receiver Overflow (Optional)		Will be Implemented in RI. Optional FC Errors
	Flow Control Protocol Errors (Optional)		
DL	LCRC Check Failure for TLP/DLLP		
	Sequence Number Check Failure		
	Replay Time-out		
	Replay Number Rollover		
	DLL Protocol Errors		
PL	Receiver/Training Errors (Optional)		

19.1.6.4.1 Role Based Error Reporting

In earlier versions of the PCI Express Specification, errors were reported by the agent that detected the error. The *PCI Express Base Specification*, Rev. 1.1 implements a role based error reporting where the response to the errors is based on the components role in the transaction. In general, errors detected in Non Posted transactions are handled by the initial requestor and the completer may optionally send an advisory message to the root complex as an ERR_COR message. Errors in posted transactions are still logged and reported by the target device.

Note: If the severity for the error is programmed to fatal in the PCI Express* Uncorrectable Error Severity register, then it is not an Advisory Non-Fatal Error and is signalled with an ERR_FATAL message. A fatal severity overrides all other Advisory Error control bits.

The following errors are considered Advisory Non-Fatal Error cases and have different handling depending based on the transaction type.

- ECRC Check Failed
- Unsupported Request (UR)
- Completer Abort (CA)
- Unexpected Completion
- Poisoned TLP Received
- Completion Timeout

Table 19-5. Advisory Error Cases

Error Type	Posted	Non Posted	Completion
ECRC Check Failed	EP does not support ECRC checking.		
Unsupported Request	Not Advisory Error - Send ERR_NONFATAL	Advisory Error - Send ERR_COR	Signaled via device driver
Completer Abort	Not Advisory Error - Send ERR_NONFATAL	Advisory Error - Send ERR_COR	Signaled via device driver
Unexpected Completion	N/A	N/A	Advisory Error - Send ERR_COR
Poisoned TLP Received	Not Advisory Error -Send ERR_NONFATAL	N/A	Not Advisory Error - Send ERR_NONFATAL
Completion Timeout	N/A	N/A	Advisory Error - send ERR_COR.

The different responses are described in detail in the following sections.

19.1.6.4.2 Malformed Packets

The following checks are made to detect malformed TLPs.

- Data Payload exceeds the length specified by the value in the Max_Payload_Size field of the Device Control register.
- The value in the length field and the actual amount of data received do not match. The value in the length field applies only to data, TLP digest is not included in the length.
- A TLP with a 1b in TD field but without a TLP digest or a TLP with a TLP digest but without a 1b in TD field
- Address/Length combination which crosses a 4K boundary.
- Receipt of Assert_INTx/Deassert_INTx messages.
- Packets having undefined Type Field

- IO and Configuration requests are considered malformed when
  - TC[2:0] /= 000b
  - Attr[1:0] /= 00b
  - Length[9:0] /= 0000000001b
  - Last DW BE[3:0] /= 0000b

When a malformed packet is detected, the packet is dropped and the error is logged. No flow control information is updated for malformed packets.

19.1.6.4.3 ECRC Check Failed

EP does not support ECRC checking.

19.1.6.4.4 Unsupported Request

Unsupported Requests are detected by the address decode and translation logic. A TLP is treated as unsupported in the following cases:

- the TLP fails to match any of the active memory or I/O windows.
- a configuration TLP that targets an invalid function number
- receipt of a Vendor_Defined Type 0 message
- a message request with an undefined or unsupported message code
- a poisoned I/O or Configuration request
- receipt of a Memory or I/O transaction while in a non-D0 power state
- receipt of a Memory Read Lock (MRdLk)

No checks are made to for address + length crossing a window boundary.

For posted transactions, this is **not** an Advisory Error and an ERR_NONFATAL is sent to the root complex.

For non-posted transactions, this is considered and Advisory Error. An ERR_COR is sent to the root complex and a completion with UR status is returned to the requestor.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an Advisory Error and an ERR_FATAL will be sent to the root complex.

19.1.6.4.5 Completer Abort

Requests that target abort or master abort on the Internal Bus are treated as a Completer Abort.

These requests must have passed the Malformed TLP checks as well as the Unsupported Request checks before they are issued on the internal bus.

For posted transactions, an ERR_NONFATAL is sent to the root complex.

For non-posted transactions, an ERR_COR is sent to the root complex and a completion with CA status is returned to the requestor.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an Advisory Error and an ERR_FATAL will be sent to the root complex.

19.1.6.4.6 Unexpected Completions

Unexpected completions occur when a completion transaction ID does not match a outstanding request. If the Requester ID of the completion matches a valid function, the error will get logged in that function. Otherwise the error will get logged against all functions.

This is an Advisory Non-Fatal Error and an ERR_COR will be sent to the root complex.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an Advisory Error and an ERR_FATAL will be sent to the root complex.

19.1.6.4.7 Poisoned TLP Received

Poisoned TLPs can be received for both Inbound Posted (Write/Message) and Inbound Completions. The two TLP types can be handled differently.

Poisoned completions are passed through to the target agent with the error bit set. The error is logged in the corresponding function. This is an Advisory Error and an ERR_COR will be issued.

Poisoned Memory Writes are passed through to the target agent with the error bit set. The error is logged in the corresponding function. This is a non-fatal error and an ERR_NONFATAL will be issued

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an non-fatal Error and an ERR_FATAL will be sent to the root complex. Auto-recovery is discouraged as the ERR_FATAL message will likely bring down the hierarchy.

19.1.6.4.8 Completion Timeout

When an out-bound non-posted request results in a completion timeout, the Advanced Error registers are updated in the corresponding function. This will be treated as an Advisory Error and an ERR_COR will be sent to the root complex.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an Advisory Error and an ERR_FATAL will be sent to the root complex. Auto-recovery is discouraged as the ERR_FATAL message will likely bring down the hierarchy.

19.1.6.4.9 Non Function Specific Errors

The PCI Express specification lists the following errors as non function specific:

- All Physical Layer errors
- All Data Link Layer errors
- These transaction Layer errors
  - ECRC Fail
  - UR, when caused by no function claiming a TLP
  - Receiver Overflow
  - Flow Control Protocol Error
  - Malformed TLP
  - Unexpected Completion, when caused by no Function claiming a Completion

On the detection of one of these errors, a multi-function device should generate at most one error reporting message of a given severity, where the message must report the Requester ID of a function of the device that is enabled to report that specific type of error. If no function is enabled to send a reporting message, the device does not send a reporting message. If all reporting-enabled functions have the same severity level set for the error, only one error message is sent. If all reporting-enabled functions do not have the same severity level set for the error, one error message for each severity level is sent. Software is responsible for scanning all functions in a multi-function device when it detects one of those errors.

19.1.7 Device Specific Error Management

EP will report device specific errors using interrupts - INTx or MSI based on the PCIe* configuration settings.

Table 19-6. Device Specific Errors

Function	Description
Function 0	EP Implements Error source and mask registers. Upon detected an asserted on an unmasked error, RI will generate either INTA or MSI.
Function [4:1[1]]	Based on GbE[1].

1. Function [4:1] (GbE) is not available in the DH89xxCL devices.

19.1.7.1 Memory Error Poisoning

The TI also tracks data errors for all Transactions that flow through the TI, including accesses to PCIe* and CSRs.

19.1.7.1.1 Memory Write Poisoning

For a PCIe* Memory Write (including Memory Writes triggered by the Ring Controller), the TI pulls data from the EP Master. If the internal bus indicates an error, then the TI logs the error, but it does not abort the Write Transaction. Instead, the TI passes along the Data Error notification along with the Data. On PCIe*, the TI will set the EP bit in the header of the packet.

19.1.8 PCIe* Request Generation

Table 19-7 shows the supported PCIe* packet types generated and accepted by the EP.

Table 19-7. Supported PCIe* Transactions

	Packet Name	Packet Description	RI Accept?	TI Generate?	Comments
TLP	MRd	Memory Read Request	Yes		
	MWr	Memory Write Request			
	Configuration Type 0 (Rd, Wr)	Type 0 Configuration Read & Write	Yes	No	EP will not generate but accept Type 0 Configuration Cycles
	Msg	Message Request without Data	No	Yes	INTx, Error etc.
	MsgD	Message Request With Data	No		
	Cpl	Completion without Data	No	Yes	For configuration requests
	CplD	Completion with Data	Yes		Read return
	MRdLk	Memory Read Request - Locked access	No		Not required for a PCIe* Device
	CplLk	Completion without Data for Locked access			
	CplDLk	Completion with Data for Locked access			
	IO (Rd, Wr)	IO Read & Write			
	Configuration Type 1 (Rd, Wr)	Type 1 Configuration Read & Write			Only for bridges/root ports
DLLP	Ack	TLP Acknowledge	Yes Terminated in Link/Phy Layer. Some functionality will require support in EP.		Link Layer functionality that is required
	Nak	TLP No Acknowledge			
	PM_Enter_L1	Power Management			EP supports L0, L1
	PM_Enter_L23				
	PM_Active_State_Request_L1				
	PM_Request_Acq				
	InitFC-(P, NP, Cpl)	Flow Control	Yes		Required for communicating flow control information between the root port & EP
	InitFC2)P, NP, CPl)				
	UpdateFC(P, NP, Cpl)				
PLP	TS1	Training Sequence 1	Yes Terminated in Link/Phy Layer. Some functionality will require support in EP.		PHY Layer functionality
	TS2	Training Sequence 2			
	SKIP	Required for compensation of clock frequency variation between transmitter and receiver			
	FTS	Fast Training Sequence Required for transition from L0s to L0			
	IDLE	Electrical Idle Ordered Sets			

The TI generates a PCIe* request header for memory based on the EP command and read completions from RI, and messages based on the EP internal interrupts and errors.

19.1.8.1 64-bit Memory Accesses

Table 19-8 and Table 19-9 shows the 3 and 4 DW PCIe* memory request header format. TI will use the 3 DW header format when upper address bits (63:32) are zero, else it will use the 4 DW header format.

Table 19-8. PCIe* Request Header Format for 32-bit Memory Addressing

Table 3:

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
R	FMT		Type					R	TC			R				TD	EP	Attr		R				Length							
Requester ID																Tag								Last DW BE				First DW BE			
Address[31:2]]																															R

Table 19-9. PCIe* Request Header Format for 64-bit Memory Addressing

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
R	FMT		Type					R	TC			R				TD	EP	Attr		R				Length							
Requester ID																Tag								Last DW BE				First DW BE			
Address[63:32]																															
Address[31:2]]																															R

TI will generate the header based on the EP command it receives and Table 19-9 shows the mapping of the EP command fields to the PCIe* command fields. Some fields are static for all PCIe* transactions initiated by the TI while others depend upon the EP command.

19.1.8.2 Completions

EP will issue completions for the IA initiated PCIe* commands:

- IA initiated memory read and configuration read commands will result in a completion with data (CplD)
- IA initiated configuration write command will result in a completion without data (Cpl)

Table 19-10. PCIe* Completion Header Format

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
R	FMT		Type					R	TC			R				TD	EP	Attr		R				Length							
Completer ID																Completion Status			BCM	Byte Count											
Requester ID																Tag								R	Lower Address						

Table 19-11. PCI Completion Request Generation (Sheet 1 of 2)

PCIe* Header Field	PCIe* Description	PCIe* Field Encoding	Generated Based on?
Length [9:0]	Indicates the Length of the data payload in DW	0x0h = 1024DW 0x1h = 1DW . . . 1FFh = 1023DW	TI will support up to 8DWs data payload for completions.
Attributes [0]	No Snoop	1 = No Snoop 0 = Snoop	Hard-code to 0 EP should never get a request with no snoop or RO bit set.
Attributes [1]	Relaxed Ordering (RO)	1=RO enabled 0=Strict Ordering	
EP	Poisoned Data	1 = Data is invalid	Based on Pushdata Error
TD	Digest	1 = Digest field is included	EP does not support ECRC so this field is set to 0 for all packets generated by the EP.
TC[2:0]	Traffic Class	000b = TC0 . . 111b = TC7	Same value as in Request Header
Fmt[1:0]	Transaction Format	00b = Cpl 10=CplD	Generated based on Type of Request that causes the completion header. MRd & CfgRd = CplD CfgWr = Cpl
Type[4:0]	Type	Hardcoded to 01010b for completion packets	
Byte Count	Remaining Byte Count until the read request is satisfied		Always set to 0. EP read returns will be always < 8DW's and the AC TI will return the data in one completion.
Byte Count Modifier	Set by PCI-X completers only		Always set to 0
Completion Status	Status of the completion packet from	000b = Successful Cmpl 001b = Unsupported Cmpl 010b =Config Req Retry 100b = Completer Abort	All encodings are possible depending upon the status of the request completion.
Completer ID	Identifies the Completer	ARI Disabled: RID[15:8] = Bus RID[7:3] = Device RID[2:0] = Function ARI Enabled: RID[15:8] = Bus RID[7:0] = Function	See Requester ID row at the end of this table for details on how Completer ID is calculated.

Table 19-11. PCI Completion Request Generation (Sheet 2 of 2)

PCIe* Header Field	PCIe* Description	PCIe* Field Encoding	Generated Based on?
Lower Address	Lower 7 bits of the first enabled byte of data returned with a read	Valid only for Completions with data and refers to the first enabled byte of data being returned by the completer	Calculated by RI for Configuration accesses based on the DW aligned address and BEs. All other MMIO accesses are either 4B or 8B aligned. There is no support for BEs for such MMIO accesses.
Tag[7:0]	Tags to identify requests		Same value as in Request Header
Requester ID[15:0]	Bus, Device, Function Number	ARI Disabled: RID[15:8] = Bus RID[7:3] = Device RID[2:0] = Function ARI Enabled: RID[15:8] = Bus RID[7:0] = Function	RID copied from the original request.

19.1.8.3 Messages

Table 19-12 shows the PCIe* message format. The EP generates INTx, error messages, and PME messages.

Table 19-12. PCIe* Request Header Format for Messages

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
R	FMT		Type					R	TC			R				TD	EP	Attr		R				Length							
Requester ID																Tag								Message Code							
Address field not used if Implicit Routing is used.																															
Address field not used if Implicit Routing is used.																															

Table 19-13. PCIe* Message Generation (Sheet 1 of 2)

PCIe* Header Field	PCIe* Description	Field Encoding	Comment
Length [9:0]	Length in DW	0x0h	Reserved for messages
Attributes [0]	No Snoop	0 = Snoop	Fixed for messages
Attributes [1]	Relaxed Ordering (RO)	0=Strict Ordering	
EP	Poisoned Data	0 = Data is not poisoned	No data packet for messages
TD	Digest	1 = Digest field is included	EP does not support ECRC so this field is set to 0 for all packets generated by the EP.

Table 19-13. PCIe* Message Generation (Sheet 2 of 2)

PCIe* Header Field	PCIe* Description	Field Encoding	Comment
TC[2:0	Traffic Class	000b = TC0	Fixed for INTx & Error messages
Fmt[1:0] Type[4:0]	Transaction Format & Type	01b & 10100b (Msg w/o Data and Terminate at Receiver)	
Message Code [7:0]	Assert_INTX	0010 0000b	Based on internal EP interrupt event or error message type. and Error Messages
	Deassert_INTX	0010 0100b	
	ERR_COR	0011 0000b	
	ERR_NONEATAL	0011 0001b	
	ERR_FATAL	0011 0011b	
Tag[7:0]	Tags to identify requests	00h	Posted Message, no tag
Requester ID[15:0]	Bus, Device, Function Number	See the Requester ID row of Table 19-11 for details on how Completer ID is calculated.	Fixed for messages

19.2 Intel® QuickAssist Technology

With the Intel® QuickAssist Integrated Accelerator (QAT) as an essential part of the EP, this allows the IA to efficiently access the QAT services through the QAT APIs.

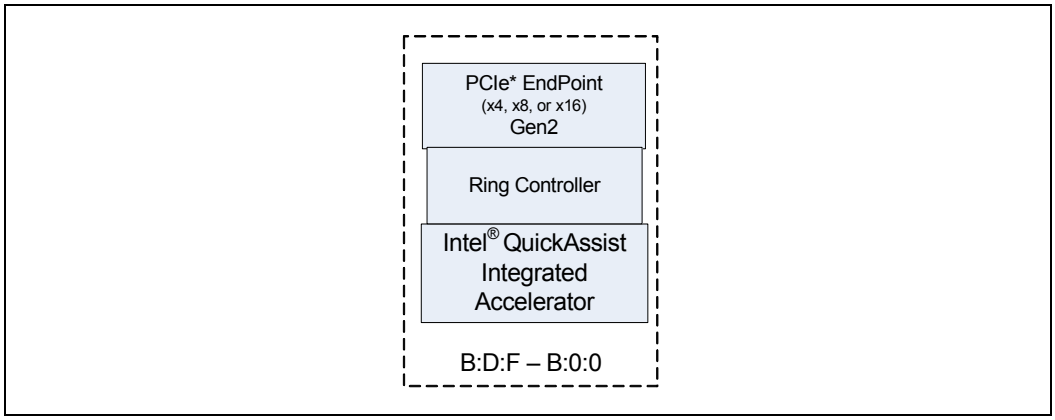
19.2.1 Features

- Symmetric Cryptographic Functions
  - — Cipher Operations
  - — Hash/Authenticate Operation
  - — Cipher-Hash Combined Operation
  - — Key Derivation Operation
  - — Random Number Generation
- Public Key Functions
  - — RSA Operation
  - — Diffie-Helman Operation
  - — Digital Signature Standard Operation
  - — Key Derivation Operation
  - — Elliptic Curve Cryptography: ECDSA* and ECDH*
  - — Random Number Generation and Prime Number Testing
- Compression/Decompression
  - — Deflate (Lempel-Ziv 77-Stac)

19.2.2 EP Intel® QuickAssist Integrated Accelerator (IQIA)

The IQIA provides acceleration functions that can be used by the IA cores. The acceleration services can be accessed by a standard PCIe* driver. To enhance communication between the IA and the IQIA, the EP implements rings via the Ring Controller to route messages between them.

Figure 19-3. EP Block Diagram



19.2.3 Ring Controller

The Ring Controller provides for communication between the IA cores and the Intel® QuickAssist Integrated Accelerator.

19.2.3.1 Features

The DH89xxCC (8900 ≤ *SKU* ≤ 8920):

- Implements 256 rings for communications between the CPU and the IQIA.
- Organized in 16 bundles of 16 rings each.
- Each Bundle supports 1 interrupt to the PCIe* Gen2 EndPoint Function 0.

The DH89xxCL (8925 ≤ *SKU* ≤ 8955):

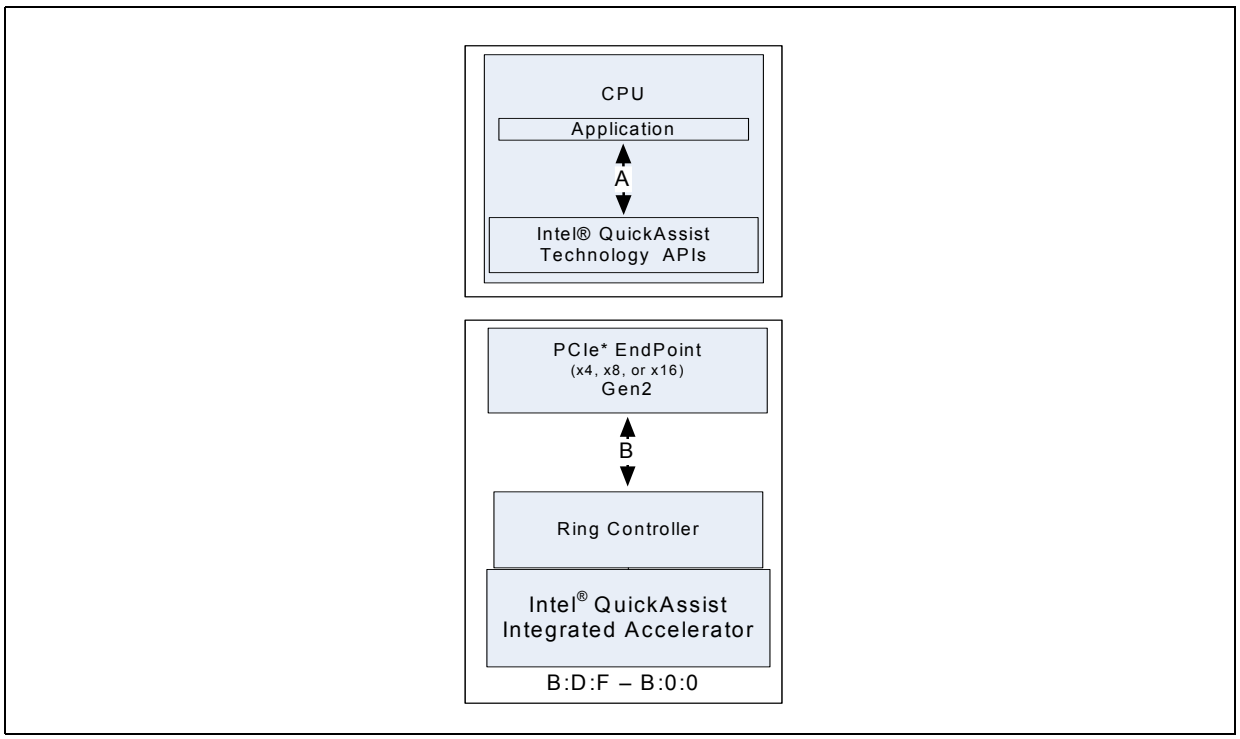
- Implements 512 rings for communications between the CPU and the IQIA.
- Organized in 32 bundles of 16 rings each.
- Each Bundle supports 1 interrupt to the PCIe* Gen2 EndPoint Function 0.

19.2.3.2 Functionality

The Intel® QuickAssist Technology is accessed from the IA CPU via Intel® QuickAssist Technology APIs. Intel provides these APIs for the QA IA Services. Interface "A" in Figure 19-4 illustrates an application accessing the IQAT through these APIs.

Interface "A" is not dependent on HW or FW, but Interface "B" is. The interface may change with subsequent releases of the driver, which is maintained by Intel.

Figure 19-4. Ring Block Diagram



Note: See the *Intel® QuickAssist Technology Programmer's Guide* for information on Intel® QuickAssist Technology APIs.

§ §

20.0 PCIe Endpoint Function 0 Registers

20.1 EP PF PCI Configuration Space

The EP Physical Function implements Type 0 PCIe configuration space that is compliant with PCIe 2.0 Specification.

Table 20-1. EP PCI Configuration Registers (Sheet 1 of 3)

	3	2	1	0	Offset[1]
Required PCI Compatible Configuration Space	Device ID		Vendor ID		00h
	Status		Command		04h
	Class Code			Revision ID	08h
	BIST	Header Type	Master Latency Timer	Cache Line Size	0Ch
	Base Address Register 1 0 (PeQATBAR)				10h
					14h
	Base Address Register 3 2 (PMISCBAR)				18h
					1Ch
	Base Address Register 5 4 (PETRINGCSRBAR)				20h
					24h
	Cardbus CIS Pointer				28h
	Subsystem ID		Subsystem Vendor ID		2Ch
	Expansion ROM Base Address				30h
	Reserved			Capabilities Pointer	34h
	Reserved				38h
	Max_Lat	Min_Gnt	Interrupt Pin	InterruptLine	3Ch
	SKU Register				40h
	RESERVED				44h - 4Bh
	SKU Register 2				4Ch
	Reserved				50-5Ch
MSI-X Cap	Message Control		Next Cap Pointer	MSI-X CAP_ID	60h
	MSI-X Table Offset and BIR				64h
	MSI-X PBA Offset and BIR				68h
PM Cap	Power Management Capabilities		Next Cap Pointer	PM CAP_ID	6Ch
	Power Management Control and Status				70h

Table 20-1. EP PCI Configuration Registers (Sheet 2 of 3)

	3	2	1	0	Offset[1]
PCIE Cap	PCI Express Capabilities Register		Next Cap Pointer	PCIe CAP_ID	74h
	Device Capabilities				78h
	Device Status		Device Control		7Ch
	Link Capability				80h
	Link Status		Link Control		84h
	Slot Capabilities (Reserved)				88h
	Slot Status (Reserved)		Slot Control (Reserved)		8Ch
	Root Capabilities (Reserved)		Root Control (Reserved)		90h
	Root Status (Reserved)				94h
	Device Capabilities 2				98h
	Device Status 2 (Reserved)		Device Control 2		9Ch
	Link Capabilities 2 (Reserved)				A0h
	Link Status 2		Link Control 2		A4h
	Slot Capabilities 2 (Reserved)				A8h
	Slot Status 2 (Reserved)		Slot Control 2 (Reserved)		ACh
MSI Cap	Message Control		Next Cap Pointer	MSI CAP_ID	B0h
	MSI Addr				B4h - BBh
	Reserved		MSI Data		BCh
	MSI Mask				C0h
	MSI pending				C4h
Reserved	Reserved				C8h - FCh
AER Configuration Space	Next Capability Pointer/Capability Version		AER Capability ID		100h
	Uncorrectable Error Status				104h
	Uncorrectable Error Mask				108h
	Uncorrectable Error Severity				10Ch
	Correctable Error Status				110h
	Correctable Error Mask				114h
	Control and Capability				118h
	Header Log				11Ch
	Header Log				120h
	Header Log				124h
	Header Log				128h
ARI Configuration Space	Next Capability Pointer/Capability Version		ARI Capability ID		138h
	ARI Control		ARI Capability		13Ch

Table 20-1. EP PCI Configuration Registers (Sheet 3 of 3)

	3	2	1	0	Offset[1]
SR-IOV Configuration Space	Next Capability Pointer/Capability Version		SRIOV Capability ID		140h
	SR-IOV Capabilities				144h
	SR-IOV Status		SR-IOV Control		148h
	Total VF's (RO)		Initial VF's (RO)		14Ch
	Reserved	Function Dependency	Num VF's (RW)		150h
	VF Stride (RO)		First VF Offset (RO)		154h
	VF Device ID(RO)		Reserved		158h
	Supported Page Sizes (RO)				15Ch
	System Page Size (RW)				160h
	VF BAR0 (RW)				164h
					168h
	VF BAR 1 (RW)				16Ch
					170h
	Reserved				174h
	Reserved				178h
	VF Migration Offset (N/A)				17Ch
Reserved	Reserved				180h - FFCh

Notes:
1. Any addresses not shown are Read-only 0.

20.2 Detailed Register Summaries

20.2.1 PCI Views

Table 20-2. Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Configuration Registers (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"PVID—PF Vendor Identification Register" on page 899	8086h
02h	03h	"PDID—PF Device Identification Register" on page 899	0434h
04h	05h	"PPCICMD—PF Device Command Register" on page 900	0000h
06h	07h	"PPCISTS—PF PCI Device Status Register" on page 901	0010h
08h	08h	"PRID—PF Revision ID Register" on page 903	10h
09h	0Bh	"PCC—PF Class Code Register" on page 903	0B4000h
0Eh	0Eh	"PHDR—PF Header Type Register" on page 904	80h
18h	1Bh	"PMISCLBAR—PF Miscellaneous Lower Base Address Register" on page 905	00000004h
1Ch	1Fh	"PMISCUBAR—PF Miscellaneous Upper Base Address Register" on page 906	00000000h

Table 20-2. Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Configuration Registers (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
20h	23h	"PETRINGCSRLBAR—PF Ring CSR Lower Base Address Register" on page 906	00000004h
2Ch	2Dh	"PSVID—PF Subsystem Vendor ID Register" on page 907	8086h
2Eh	2Fh	"PSID—PF Subsystem ID Register" on page 907	0000h
34h	34h	"PCP—PF Capabilities Pointer Register" on page 908	B0h
3Ch	3Ch	"PIRQL—PF Interrupt Line Register" on page 908	00h
3Dh	3Dh	"PIRQP—PF Interrupt Pin Register" on page 909	01h
60h	60h	"PMSI-X—PF Message Signalled Interrupt X Capability ID Register" on page 911	11h
61h	61h	"PMSIXNCP—PF MSIX Next Capability Pointer Register" on page 911	6Ch
62h	63h	"PMSIXCNTL—PF Message Signalled Interrupt X Control Register" on page 912	0010h
64h	67h	"PMSIXTBIR—PF MSI-X Table Offset & Table BIR Register" on page 912	0001B002h
68h	6Bh	"PMSIXPBABIR—PF MSI-X Pending Bit Array & BIR Offset Register" on page 913	0001B802h
6Ch	6Ch	"PPMCAP—PF Power Management Capabilities ID Register" on page 913	01h
6Dh	6Dh	"PPMCP—PF Power Management Next Capability Pointer Register" on page 914	74h
6Eh	6Fh	"PPMC—PF Power Management Capabilities Register" on page 914	0023h
70h	73h	"PPMCSR—PF Power Management Control and Status Register" on page 915	00000000h
74h	74h	"PPCID—PF PCI Express Capability Register" on page 917	10h
75h	75h	"PPCP—PF PCI Express Next Capability Pointer Register" on page 917	0h
76h	77h	"PPCR—PF PCI Express Capabilities Register" on page 918	0002h
78h	7Bh	"PPDCAP—PF PCI Express Device Capabilities Register" on page 918	10008041h
7Ch	7Dh	"PPDCNTL—PF PCI Express Device Control Register" on page 920	2810h
7Eh	7Fh	"PPDSTAT—PF PCI Express Device Status Register" on page 921	0000h
80h	83h	"PLCAPR—PF Link Capabilities Register" on page 922	3B502h
84h	85h	"PLCNTLR—PF Link Control Register" on page 924	0000h
86h	87h	"PLSR—PF Link Status Register" on page 926	0102h
98h	9Bh	"PDCAPR2—PF Device Capabilities 2 Register" on page 927	00000012h
9Ch	9Dh	"PDCNTR2—PF Device Control 2 Register" on page 928	0000h
A4h	A5h	"PLCNTLR2—PF Link Control 2 Register" on page 929	0002h
A6h	A7h	"PLSR2—PF Link Status 2 Register" on page 931	0000h
B0h	B0h	"PMSICID—PF Message Signalled Interrupt Capability ID Register" on page 931	05h
B1h	B1h	"PMSINCP—PF Message Signalled Interrupt Next Capability Pointer Register" on page 932	60h
B2h	B3h	"PMSICTL—PF Message Signalled Interrupt Control Register" on page 932	0180h
B4h	B7h	"PMSILADDR—PF Message Signalled Interrupt Lower Address Register" on page 933	00000000h
B8h	BBh	"PMSIUADDR—PF Message Signalled Interrupt Upper Address Register" on page 933	00000000h
BCh	BDh	"PMSIDATA—PF Message Signalled Interrupt Data Register" on page 934	0000h
C0h	C3h	"PMSIMSK - PF Message Signalled Interrupt Mask Register" on page 934	00000000h
C4h	C7h	"PMSIPND—PF Message Signalled Interrupt Pending Register" on page 935	00000000h
100h	103h	"PPCIEAERCAPID—PF PCI Express AER Capability ID Register" on page 935	13810001h

Table 20-2. Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Configuration Registers (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
104h	107h	"PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register" on page 936	0h
108h	10Bh	"PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" on page 937	0h
10Ch	10Fh	"PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" on page 938	00062030h
110h	113h	"PPAERCS—PF PCI Express AER Correctable Error Register" on page 939	00h
114h	117h	"PPAERCM—PF PCI Express AER Correctable Error Mask Register" on page 940	2000h
118h	11Bh	"PPAERCTLCAP—PF PCI Express AER Control and Capability Register" on page 941	0h
11Ch	11Fh	"PPAERHDRLOG0—PF PCI Express AER Header Log 0 Register" on page 941	0h
120h	123h	"PPAERHDRLOG1—PF PCI Express AER Header Log 1 Register" on page 942	0h
124h	127h	"PPAERHDRLOG2—PF PCI Express AER Header Log 2 Register" on page 942	0h
128h	12Bh	"PPAERHDRLOG3—PF PCI Express AER Header Log 3 Register" on page 943	0h
138h	13Bh	"PARIDHDR—PF Alternative Routing ID Capability Header" on page 944	1401000Eh
13Ch	13Dh	"PFARICAP—PF ARI Capabilities Register" on page 944	0100h
13Eh	13Fh	"PARIDCTL—PF Alternative Routing ID Control Register" on page 945	00000000h
140h	141h	"PSRIOVCAPID—PF SR IOV Capability ID Register" on page 945	10h
142h	143h	"PSRIOVCVNC—PF SRIOV Capability Version and Next Capability Pointer Register" on page 946	1h
144h	147h	"PSRIOVCAP—PF SRIOV Capabilities Register" on page 947	00000000h
148h	14Bh	"PSRIOVCS—PF SRIOV Control and Status Register" on page 947	00000000h
14Ch	14Fh	"PSRIOVMTOTINI—PF SRIOV Initial and Total VFs Register" on page 948	00100010h
150h	153h	"PSRIOVNUMVF—PF SRIOV Number of VFs Register" on page 948	00000000h
154h	155h	"PSRIOVFVFO—PF SRIOV First VF Offset Register" on page 949	0008h
156h	157h	"PSRIOVVFS—PF SRIOV VF Stride Register" on page 949	0001h
158h	15Bh	"PSRIOVFDID—PF SRIOV VF Device ID Register" on page 950	04420000h
15Ch	15Fh	"PSRIOVPAGESIZE—PF SRIOV Supported Page Size Register" on page 950	00000553h
160h	163h	"PSRIOVSYSPS—PF SRIOV System Page Size Register" on page 951	00000001h
164h	167h	"PSRIOVLBAR0—SRIOV Lower BAR0 Register" on page 952	00000004h
168h	16Bh	"PSRIOVUBAR0—SRIOV Upper BAR0 Register" on page 953	00000000h
16Ch	16Fh	"PSRIOVLBAR1—SRIOV Lower BAR1 Register" on page 954	00000004h
170h	173h	"PSRIOVUBAR1—SRIOV Upper BAR1 Register" on page 955	00000000h
17Ch	17Fh	"PSRIOVVFMA—PF SRIOV VF Migration Array Register" on page 955	00000000h

Table 20-3. Bus M, Device 0, Function 0: Summary of PCIe Intel® QuickAssist Registers Mapped Through PMISCRBAR+1A000h Memory BAR

Offset Start	Offset End	Register ID - Description	Default Value
008h	00Bh	"ERRSOU2—Error Source Register 2" on page 985	00000000h
018h	01Bh	"ERRMSK2—Error Source Mask Register 2" on page 985	00000000h
024h	027h	"SINTPF—Signal Raw PF Interrupt Register" on page 986	00000000h
028h	02Bh	"SMIAPF—Signal IA PF Interrupt Mask Register" on page 986	00000000h
03Ch	03Fh	"GBECFGMMIOV - GBE Configuration and MMIO Valid Register" on page 987	00000000h
040h / at 14h	043h / at 14h	"SINTGBE[0:3]—Signal Raw PF Interrupt for GBE Register" on page 988	00000000h
044h / at 14h	047h / at 14h	"SMIAGBE[0:3]—Signal IA PF Interrupt Mask GBE Register" on page 988	000003FFh

Table 20-4. Bus M, Function 0: Summary of PCIe Intel® QuickAssist Registers Mapped Through PMISCSBAR+1A000h Memory BAR

Offset Start	Offset End	Register ID - Description	Default Value
03Ch	03Fh	"GBECFGMMIOV - GBE Configuration and MMIO Valid Register" on page 987	00000000h

Table 20-5. Bus M, Function 8+Index 1: Summary of PCIe Intel® QuickAssist Configuration Registers (Sheet 1 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"VVID[0:15]—VF Vendor Identification Register" on page 956	FFFFh
02h	03h	"VDID[0:15]—VF Device Identification Register" on page 956	FFFFh
04h	05h	"VPCICMD[0:15]—VF Device Command Register" on page 957	0000h
06h	07h	"VPCISTS[0:15]—VF PCI Device Status Register" on page 958	0010h
08h	08h	"VRID[0:15]—VF Revision ID Register" on page 959	10h
09h	0Bh	"VCC[0:15]—VF Class Code Register" on page 959	0B4000h
0Eh	0Eh	"VHDR[0:15]—VF Header Type Register" on page 959	00h
2Ch	2Dh	"VSVID[0:15]—VF Subsystem Vendor ID Register" on page 960	8086h
2Eh	2Fh	"VSID[0:15]—VF Subsystem ID Register" on page 960	0000h
34h	34h	"VCP[0:15]—VF Capabilities Pointer Register" on page 961	90h
3Ch	3Ch	"VIRQL[0:15]—VF Interrupt Line Register" on page 961	00h
3Dh	3Dh	"VIRQP[0:15]—VF Interrupt Pin Register" on page 961	00h
50h	50h	"VPCID[0:15]—VF PCI Express Capability ID Register" on page 962	10h
51h	51h	"VPCP[0:15]—VF PCI Express Next Capability Pointer Register" on page 962	0h
52h	53h	"VPCR[0:15]—VF PCI Express Capabilities Register" on page 963	002h
54h	54h	"VPDCAP[0:15]—VF PCI Express Device Capabilities Register" on page 963	10008041h

Table 20-5. Bus M, Function 8+Index 1: Summary of PCIe Intel® QuickAssist Configuration Registers (Sheet 2 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
58h	59h	"VPDC[0:15]—VF PCI Express Device Control Register" on page 964	00h
5Ah	5Bh	"VPDS[0:15]—VF PCI Express Device Status Register" on page 965	0000h
5Ch	5Fh	"VLCR[0:15]—VF Link Capabilities Register" on page 966	3B502h
60h	61h	"VLCNTLR[0:15]—VF Link Control Register" on page 968	0000h
62h	63h	"VLSR[0:15]—VF Link Status Register" on page 969	0000h
74h	77h	"DCAPR2[0:15]—Device Capabilities 2 Register" on page 969	00000012h
90h	90h	"VMSICID[0:15]—Message Signalled Interrupt Capability ID Register" on page 970	05h
91h	91h	"VMSINCP[0:15]—Message Signalled Interrupt Next Capability Pointer Register" on page 970	00h
92h	93h	"VMSICTL[0:15]—Message Signalled Interrupt Control Register" on page 971	0180h
94h	97h	"VMSILADDR[0:15]—Message Signalled Interrupt Lower Address Register" on page 971	00000000h
98h	9Bh	"VMSIUADDR[0:15]—Message Signalled Interrupt Upper Address Register" on page 972	00000000h
9Ch	9Dh	"VMSIDATA[0:15]—Message Signalled Interrupt Data Register" on page 972	0000h
100h	103h	"VPCIEAERCAPID[0:15]—VF PCI Express AER Capability ID Register" on page 973	13810001h
104h	107h	"VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register" on page 974	0h
108h	10Bh	"VPAERUCM[0:15]—VF PCI Express AER Uncorrectable Error Mask Register" on page 975	0h
10Ch	10Fh	"VPAERUCSEV[0:15]—VF PCI Express AER Uncorrectable Error Severity Register" on page 977	00h
110h	113h	"VPAERCS[0:15]—VF PCI Express AER Correctable Error Status Register" on page 978	0h
114h	117h	"VPAERCM[0:15]—VF PCI Express AER Correctable Error Mask Register" on page 979	0h
118h	11Bh	"VPAERCTLCAP[0:15]—VF PCI Express AER Control and Capability Register" on page 980	0h
11Ch	11Fh	"VPAERHDRLOG0[0:15]—VF PCI Express AER Header Log 0 Register" on page 981	0h
120h	123h	"VPAERHDRLOG1[0:15]—VF PCI Express AER Header Log 1 Register" on page 982	0h
124h	127h	"VPAERHDRLOG2[0:15]—VF PCI Express AER Header Log 2 Register" on page 982	0h
128h	12Bh	"VPAERHDRLOG3[0:15]—VF PCI Express AER Header Log 3 Register" on page 983	0h
138h	13Bh	"VARIDHDR[0:15]—VF Alternative Routing ID Capability Header" on page 983	1000Eh
13Ch	13Dh	"VFARICAP[0:15]—VF ARI Capabilities Register" on page 984	00000000h
13Eh	13Fh	"VARIDCTL[0:15]— VF Alternative Routing ID Control Register" on page 984	00000000h

20.2.2 PCI Standard Header Registers

20.2.2.1 PVID—PF Vendor Identification Register

The VID Register contains the vendor identification number. This 16-bit register combined with the Device Identification Register uniquely identifies any PCI device. Writes to this register have no effect.

Table 20-6. PVID—PF Vendor Identification Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 00h **Offset End:** 01h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	**Vendor Identification:** This register field contains the PCI standard identification for Intel, 8086h.		8086h	RO

20.2.2.2 PDID—PF Device Identification Register

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

Table 20-7. PDID—PF Device Identification Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 02h **Offset End:** 03h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0434h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	**Device Identification Number:** PCH ID = 0434h.		0434h	RO

20.2.2.3 PPCICMD—PF Device Command Register

Table 20-8. PPCICMD—PF Device Command Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 04h **Offset End:** 05h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		0h	RV
10	INTD	**Interrupt Disable:** Setting this bit disables generation of INTX messages by the EP.. Default value is 0 which enables the INTX message generation.		0h	RW
09	FBTB	**Fast Back-to-Back Enable:** EP does not implement this functionality and it is not applicable to PCIe devices. The bit is hardwired to 0.		0h	RO
08	SER	**SERR# Enable:** When set, this bit enables the non-fatal and fatal errors detected by the EP to be reported to the RC. The error reporting can also be enabled via the PCIe specific bits in the PCIe device control register (Section 20.2.5.5, "PPDCNTL—PF PCI Express Device Control Register") The default value of this bit is 0.		0h	RW
07	Reserved	Reserved/Does not apply to PCIe.		0h	RV
06	PER	**Parity Error Enable:** Controls the setting of the Master Data Parity Error bit in the Device Status Register (Section 20.2.5.6, "PPDSTAT—PF PCI Express Device Status Register") The Master Data Parity Error bit is set by the EP if its Parity Error Enable bit is set and either of the following two conditions occurs: • If the EP receives a poisoned Completion from the RC • If the EP poisons a write request. If the Parity Error Enable bit is cleared, the Master Data Parity Error status bit is never set The default value of this bit is 0.		0h	RW
05	VPS	**VGA Palette Snoop Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO
04	MWE	**Memory Write and Invalidate Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO
03	SS	**Special Cycle Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO

Table 20-8. PPCICMD—PF Device Command Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 04h **Offset End:** 05h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BM	**Bus Master Enable:** Controls the ability of EP to issue Memory Read/Write Requests. Clearing (0) this bit prevents EP from issuing any Memory Requests. Because MSIs are in-band memory writes, disabling the bus master enable bit disables MSI as well. PCIe messages are not affected by this bit. See Section 23.3.2, "BME (Bus Master Enable)" for more details on how the EP handles internal commands when this bit is clear.		0h	RW
01	MEM	**Memory Space Enable**: Setting this bit enables access to the memory regions the device claims through its BARs. EP will return "unsupported request" completion status & error message in response to memory transactions it receives when this bit is clear.		0h	RW
00	IO	**I/O Space Enable:** The device does not implement this functionality since it claims no I/O regions. The bit is hardwired to 0.		0h	RO

20.2.2.4 PPCISTS—PF Device Status Register

Table 20-9. PPCISTS—PF PCI Device Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 06h **Offset End:** 07h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	**Detected Parity Error:** This bit is set by EP whenever it receives a Poisoned TLP, regardless of the state the Parity Error Enable bit in the Command register. Default value of this field is 0.		0h	RW1C
14	SSE	**Signaled System Error:** This bit is set by the EP when it sends a ERR_FATAL or ERR_NONEATAL message and the SERR bit in the Device Command register bit is set. Default value of this field is 0.		0h	RW1C
13	RMA	**Received Master Abort Status:** This bit is set when EP, as a Requestor receives a Completion with Unsupported Request Completion Status. Default value of this field is 0.		0h	RW1C
12	RTA	**Received Target Abort Status:** This bit is set when EP, as a Requestor receives a Completion with Completer Abort Completion Status. Default value of this field is 0.		0h	RW1C

Table 20-9. PPCISTS—PF PCI Device Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 06h **Offset End:** 07h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	STA	**Signaled Target Abort Status:** This bit is set when EP completes a Request using Completer Abort Completion Status. Default value of this field is 0.		0h	RW1C
10 :09	DST	**DEVSEL Timing:** Does not apply to PCI Express. These bits are hardwired to 0.		00b	RO
08	MDPE	**Master Data Parity Error Detected:** This bit is set by EP, as a Requestor if the Parity Error Enable bit in the Command register is 1b and either of the following two conditions occurs: • Requestor receives a Completion marked poisoned • Requestor detects a parity error in the inbound completion data fifo. • Requestor poisons a write Request If the Parity Error Enable bit is 0b, this bit is never set. Default value of this field is 0.		0h	RW1C
07	FB2B	**Fast Back-to-Back Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
06	Reserved	Reserved		0h	RV
05	MC66	**66 MHz Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
04	CL	**Capabilities List:** This bit is hardwired to 1 to indicate that EP has a capabilities list.		1h	RO
03	IS	**Interrupt Status:** Indicates that the EP has transmitted a INTX message and is awaiting servicing. This bit does not includes MSI's generated by the EP.		0h	RO
02 :00	Reserved	Reserved		0h	RV

20.2.2.4.1 PRID—PF Revision ID Register

The value of this register indicates the chip stepping. It is hardwired on chip and reflects the latest revision.

Table 20-10. PRID—PF Revision ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 08h **Offset End:** 08h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :4	RIDU	**Major Revision:** Steppings which require all masks to be regenerated. 00b: A stepping 01b: B stepping 10b: C stepping (current stepping) 11b: D stepping		0010b	RO
03 :0	RIDL	**Minor Revision:** Incremented for each stepping which does not modify all masks. Reset for each major revision. 00b: x0 stepping 01b: x1 stepping (current stepping) 10b: x2 stepping 11b: x3 stepping		0001b	RO

20.2.2.5 PCC—PF Class Code Register

Table 20-11. PCC—PF Class Code Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 09h **Offset End:** 0Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 09h **Offset End:** 0Bh	
Size: 24 bit	**Default:** 0B4000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :0	CC	**Class Code:** This value indicates the base class, subclass, and interface. 0B4000h = Base class: Processor, Sub-class Co-processor, no specific register level programming interfaces are defined.		0B4000h	RWOS

20.2.2.6 PHDR—PF Header Type Register

Table 20-12. PHDR—PF Header Type Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 0Eh **Offset End:** 0Eh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 80h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HDR	**PCI Header Type:** The header type of the EP device. 00h = multi-function device with standard header layout.		80h	RO

20.2.2.7 PeQATLBAR—PF QAT Lower Base Address Register

This BAR points to the EP Intel® QuickAssist Technology. This BAR defined a 512 KB window.

Table 20-13. PeSRAMLBAR—PF Memory Lower Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 10h **Offset End:** 13h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 0000000Ch			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	ADDR	**Lower Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
18 :04	ZERO	**Lower Bits:** Hardwired to 0 (512kB region) - the size of this window is equal to QAT memory capacity		0h	RO
03	PREF	**Prefetchable:** Hardwired to 1 to indicate that the region is prefetchable. 0: non-prefetchable 1: prefetchable		1b	RO
02 :01	TYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

20.2.2.8 QATUBAR—PF QAT Upper Base Address Register

This BAR points to the EP Intel® QuickAssist Technology. This BAR defined a 512 KB window.

Table 20-14. PeQATUBAR—PF QAT Upper Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 14h **Offset End:** 17h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW

20.2.2.9 PMISCLBAR—PF Miscellaneous Lower Base Address Register

This BAR is used to access EP and MSI-X Registers. For the EP registers, see Section 20.4, "EP Memory Mapped Registers". The MSI-X tables are also included in the EP register section, starting from Section 20.4.2.5, "GBECFGMMIOV—GBE Configuration and MMIO Valid Register" thru Section 29.4.3.5, "PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register"

Table 20-15. PMISCLBAR—PF Miscellaneous Lower Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 18h **Offset End:** 1Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :17	ADDR	**Lower Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
16 :04	ZERO	**Lower Bits:** Hardwired to 0 (128KB region).		0h	RO
03	PREF	**Prefetchable:** Hardwired to 0 to indicate that the region is non-prefetchable. 0: non-prefetchable 1: prefetchable		0b	RO
02 :01	TYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

20.2.2.10 PMISCUBAR—PF Miscellaneous Upper Base Address Register

This BAR is used to access the EP MMIO space.

Table 20-16. PMISCUBAR—PF Miscellaneous Upper Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 1Ch **Offset End:** 1Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW

20.2.2.11 PETRINGCSRLBAR—PF Ring CSR Lower Base Address Register

This BAR points to the Ring Controller rings.

20.2.2.12 PETRINGCSRUBAR—PF Ring CSR Upper Base Address Register

Table 20-17. PETRINGCSRLBAR—PF Ring CSR Lower Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 20h **Offset End:** 23h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	ADDR	**Lower Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
13 :04	ZERO	**Lower Bits:** Hardwired to 0 (16KB region).		0h	RO
03	PREF	**Prefetchable:** Hardwired to 0 to indicate that the region is non-prefetchable. 0: non-prefetchable 1: prefetchable		0b	RO
02 :01	TYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

This BAR points to the Ring Controller rings.

20.2.2.13 PSVID—PF Subsystem Vendor ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

Table 20-18. PSVID—PF Subsystem Vendor ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 2Ch **Offset End:** 2Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	**Subsystem Vendor ID:** This field is hardwired to ID assigned to Intel.		8086h	RWOS

20.2.2.14 PSID—PF Subsystem ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

Table 20-19. PSID—PF Subsystem ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 2Eh **Offset End:** 2Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	**Subsystem ID:** Vendor supplied device ID. Default is 0h.		0h	RWOS

20.2.2.15 PCP—PF Capabilities Pointer Register

The Capabilities Pointer Register provides the offset in configuration space to the location where the first set of capabilities registers is located.

Table 20-20. PCP—PF Capabilities Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 34h **Offset End:** 34h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** B0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	**Pointer to First Capability Structure:** Value points to the configuration space offset of the first capability structure (MSI).		B0h	RO

20.2.2.16 PIRQL—PF Interrupt Line Register

Table 20-21. PIRQL—PF Interrupt Line Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 3Ch **Offset End:** 3Ch	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQL	**Interrupt Line:** BIOS writes the interrupt routing information to this register to indicate which input of the interrupt controller this device is connected to. The device itself does not use this information.		0h	RW

20.2.2.17 PIRQP—PF Interrupt Pin Register

Table 20-22. PIRQP—PF Interrupt Pin Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 3Dh **Offset End:** 3Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQP	**Interrupt Pin:** Set to 01h to indicate that EP uses INTA# as its interrupt pin.		01h	RO

20.2.2.18 SKU: SKU Register

For the DH89XXCL devices refer to bit field [21:20] for SKU identification.

For the DH89XXCC devices, this SKU Register (040h) in conjunction with SKU Register 2 (04Ch) determines the PCH SKU. Use the following procedure to determine the SKU:

1. Check the "Bus:Device:Function: M:0:0:08h" register to determine if A0, B0, or C0 stepping.

2. IF A0 then it is SKU4

   IF NOT A0 AND bits[9:0] = 3FFh THEN it is SKU1

   ELSE go to SKU Register 2 (04Ch).

Table 20-23. SKU Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 040h **Offset End:** 043h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 040h **Offset End:** 043h	
Size: 32 bit	**Default:**	xxxxxxxxh		**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved	N	xh	RV
21 :20	CLSKU	Identify the DH89XXCL Product SKU: 00b => DH8925CL 01b => DH8950CL 10b => Reserved 11b => DH8955CL	N	xh	RO
19 :10	Reserved	Reserved	N		RV
09 :00	SKU	SKU Register calculation value	N	xh	RO

20.2.2.19 SKU Register 2

This SKU Register 2 (04Ch) in conjunction with SKU Register (040h) determines the PCH SKU. Use the following procedure to determine the SKU (Complete steps 1 & 2 for SKU Register (040h) prior to attempting the following steps):

3. (For non-A0 steppings of the PCH)

   IF bits[22:19] =0111b it is SKU4

   IF bits[22:19] = 0110 AND bits[17:16] = 01 it is SKU3

   ELSE it is SKU2

Table 20-24. SKU Register 2

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 04Ch **Offset End:** 04Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 04Ch **Offset End:** 04Fh	
Size: 32 bit	**Default:**	xxxxxxxxh		**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved	Reserved	N	xh	RV
22 :19	SKUR2	SKU Register 2A calculation values	N	xh	RO
18	Reserved	Reserved	N	xh	RV
17 :16	SKUR2B	SKU Register 2B calculation value	N	xh	RO
15 :00	Reserved	Reserved	N	xh	RV

20.2.3 MSI-X Capability Structure

20.2.3.1 PMSI-X—PF Message Signalled Interrupt X Capability ID Register

MSI-X Capability ID Register indicates that the device is capable of generating MSI-X Interrupts.

Table 20-25. PMSI-X—PF Message Signalled Interrupt X Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 60h **Offset End:** 60h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 60h **Offset End:** 60h	
Size: 8 bit	**Default:** 11h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MSIX	**Capability ID:** PCI SIG assigned capability record ID (11h, MSI-X capability). 11h identifies the EP as a device that is MSI-X capable.		11h	RO

20.2.3.2 PMSIXNCP—PF MSIX Next Capability Pointer Register

Table 20-26. PMSIXNCP—PF MSIX Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 61h **Offset End:** 61h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 61h **Offset End:** 61h	
Size: 8 bit	**Default:** 6Ch			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Power Management Capability		6Ch	RO

20.2.3.3 PMSIXCNTL—PF Message Signalled Interrupt X Control Register

Table 20-27. PMSIXCNTL—PF Message Signalled Interrupt X Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 62h **Offset End:** 63h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 62h **Offset End:** 63h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	MSIXEN	MSI-X Enable: This bit enables the EP to generates interrupts using the MSI-X tables instead of the legacy INTx messages. When this bit is set to 1, the EP will not generate INTx messages and must use the MSI-X to signal interrupts. The device driver should not clear this bit to mask interrupts. This bit will be set by the system PCI device manager.		0h	RW
14	FM	Function Mask: This bit controls the masking of all vectors implemented in the EP. When this bit is 0, each vectors mask bit determines whether the vector is masked or not. Whent this bit is 1, all vecotrs are masked regardless of the per vector masking bit.		0h	RW
13 :11	Reserved	Always returns 0's when read.		0h	RO
10 :00	TS	MSI-X Table Size: Number of vectors (encoded as N-1) supported by the EP. The EP supports 17 vectors.		10h	RO

20.2.3.4 PMSIXTBIR—PF MSI-X Table Offset & Table BIR Register

Table 20-28. PMSIXTBIR—PF MSI-X Table Offset & Table BIR Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 64h **Offset End:** 67h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 0001B002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	TO	Table Offset: Offset to a location in one of the EP's BARs (indicated by TBIR) that points to the location of the base of the MSI-X Table. The EP MSI-X Table maps to an offset of 108KB in PMISCBAR.		00000000000000011011000000000b	RO
2 :0	TBIR	Table BAR Indicator Register: The BIR points to the EP PMISCBAR register (18h)		010b	RO

20.2.3.5 PMSIXPBABIR—PF MSI-X Pending Bit Array & BIR Offset Register

Table 20-29. PMSIXPBABIR—PF MSI-X Pending Bit Array & BIR Offset Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 68h **Offset End:** 6Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 68h **Offset End:** 6Bh	
Size: 32 bit	**Default:** 0001B802h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	PBAO	Pending Bit Array Offset: Offset to a location in one of the EP BAR's (indicated by PBABIR) that points to the location of the base of the MSI-X PBA. The EP MSI-X PBA maps to an offset of 110KB in PMISCBAR.		000000000000 000110111000 00000000b	RO
2 :00	PBABIR	Pending Bit Array BAR Indicator Register: The PBABIR points to the EP PMISCBAR register (18h)		010b	RO

20.2.4 Power Management Capability Structure

20.2.4.1 PPMCAP—PF Power Management Capabilities ID Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 20-30. PPMCAP—PF Power Management Capabilities ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 6Ch **Offset End:** 6Ch	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 6Ch **Offset End:** 6Ch	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PMID	**Capability ID:** PCI SIG assigned capability record ID (01h, Power Management capability)		01h	RO

20.2.4.2 PPMCP—PF Power Management Next Capability Pointer Register

Table 20-31. PPMCP—PF Power Management Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 6Dh **Offset End:** 6Dh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 6Dh **Offset End:** 6Dh	
Size: 8 bit	**Default:** 74h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PMCP	**Next Capability Pointer:** PCI Express Capability.		74h	RO

20.2.4.3 PPMC—PF Power Management Capabilities Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 20-32. PPMC—PF Power Management Capabilities Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 6Eh **Offset End:** 6Fh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 6Eh **Offset End:** 6Fh	
Size: 16 bit	**Default:** 0023h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	PME	**PME Support**: Set to indicate that the EP will not assert PME# in any state.		00000b	RO
10	D2	**D2 Support**. EP does not support D2 state.		0b	RO
9	D1	**D1 Support**. EP does not support D1 state.		0b	RO
8 :6	AC	**Aux current**: Not relevant for EP.		000b	RO
5	DSI	**Device specific initialization**: A one in this bit indicates that immediately after entry into the D0 Uninitialized state, the function requires additional configuration above and beyond setup of its PCI configuration Header registers before the Class driver can use the function. For the EP this bit is set to 1.		1b	RO
4	Reserved	Reserved		0b	RV
3	PMC	**PME clock**. Does not apply to PCIe.		0b	RO
2 :0	PMV	**Version**. This field is set to 3 to indicate that the EP is compliant with the PCI bus Power Management Interface Specification Rev 1.2.		011b	RO

20.2.4.4 PPMCSR—PF Power Management Control and Status Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 20-33. PPMCSR—PF Power Management Control and Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 70h **Offset End:** 73h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved:Not Supported for EP.		0000h	RO
15	PMS	**PME Status**: Hardwired to 0 since EP will not generate PME.		0b	RO
14 :13	DSC	**Data Scale**: Set to 0 because the EP does not implement the Data register. If the Data register has not been implemented, this field must return “00b” when the PMCSR is read.		00b	RO
12 :9	DSEL	**Data Select**: Set to 0 for EP because the EP does not implement the Data register (The Data register is an optional, 8-bit read-only register that provides a mechanism for the function to report state dependent operating data such as power consumed or heat dissipation) If the Data register is not implemented, this field should be read only and return “0000b” when the PMCSR is read.		0000b	RO
8	PME	**PME Enable**: Hardwired to 0 since EP does not generate a PME.		0b	RO
7 :4	Reserved	Reserved		0000b	RV

Table 20-33. PPMCSR—PF Power Management Control and Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 70h **Offset End:** 73h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	NSR	**No_Soft_Reset**: Set to 0. The EP will not maintain functional context in the D3_hot state. So when the EP is transitioned from D3_hot to D0 it will be in the D0_uniniitialized state and therefore would require software to initialize it before it can transition to the D0_initialized state. When set (“1”), this bit indicates that devices transitioning from D3hot to D0 because of Power State commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the D3_hot to the D0_Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits. When clear (“0”), devices do perform an internal reset upon transitioning from D3_hot to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from the D3_hot to the D0 state, full reinitialization sequence is needed to return the device to D0_initialized state		0b	RO
2	Reserved	Reserved		0b	RO
1 :0	PS	**Power State**. This field is used to determine the current power state of EP and to set a new power state. • 00: D0 • 01: D1 (Not supported by EP, ignore writes with this value) • 10: D2 (Not supported by EP, ignore writes with this value) • 11: D3_hot If software selects a Power state that is not supported by the EP (D2/D1), the writes must complete normally on PCIe, but the write data is discarded and no state change occurs.		00b	RW

20.2.5 PCI Express Capability Structure

20.2.5.1 PPCID—PF PCI Express Capability ID Register

The PCI Express Capability List register enumerates the PCI Express Capability structure in the PCI 3.0 configuration space capability list.

Table 20-34. PPCID—PF PCI Express Capability Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 74h **Offset End:** 74h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 74h **Offset End:** 74h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIECID	**Capability ID:** PCI SIG assigned capability record ID (10h, PCI Express capability)		10h	RO

20.2.5.2 PPCP—PF PCI Express Next Capability Pointer Register

Table 20-35. PPCP—PF PCI Express Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 75h **Offset End:** 75h	
View: PCI PF	**BAR:** Configuration	**Bus:Device:Function:** M:0		**Offset Start:** 75h **Offset End:** 75h	
Size: 8 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIENP	**Next Capability Pointer:** Last Capability.		00h	RO

20.2.5.3 PPCR—PF PCI Express Capabilities Register

Table 20-36. PPCR—PF PCI Express Capabilities Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 76h **Offset End:** 77h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 76h **Offset End:** 77h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved		00b	RO
13 :9	IMN	**Interrupt Message Number:** This field indicates which MSI vector is used for the interrupt message generated in association with the status bits in either the Slot Status field of this capability structure (applies to only RC or Switch) Not applicable to EP.		00000b	RO
8 8	SI	**Slot Implemented:** This bit when set indicates that the PCI Express Link associated with this port is connected to a slot. Hardwired to 0 for EP.		0b	RO
7 :4	DPT	**Device/Port Type:** Indicates the type of PCI Express logical device. Hardwired to 0000b (PCIe Endpoint)		0000b	RO
3 :0	CV	**Capability Version:** Indicates PCI-SIG defined PCI Express capability structure version number. EP is PCIe 2.0 Specification Compliant.		0010b	RO

20.2.5.4 PPDCAP—PF PCI Express Device Capabilities Register

Table 20-37. PPDCAP—PF PCI Express Device Capabilities Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 78h **Offset End:** 7Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved	Reserved		0h	RO
28	FLR	**Function level reset**: When set indicates that the device supports the FLR feature.		1b	RO
27 :26	CSPS	**Captured Slot Power Limit Scale:** Does not apply to EP.		00b	RO
25 :18	CSPV	**Captured Slot Power Limit Value:** Does not apply to EP.		0h	RO
17 :16	Reserved	Reserved		0h	RO
15	RBEP	**Role-Based Error Reporting:** Indicates that EP conforms to Role based error reporting ECN for PCIe 1.0a and which was subsequently rolled in PCIe 1.1 and future revisions.		1b	RO

Table 20-37. PPDCAP—PF PCI Express Device Capabilities Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 78h **Offset End:** 7Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
14 :12	ATTN	**Attention Button/Indicator Present and Power Indicator Present**. None of these are implemented in the EP.		000b	RO
11 :9	EL1L	**Endpoint L1 Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. Hardcoded to the lowest value of 1us.		000b	R0
8 :6	EL0L	**Endpoint L0s Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. This value is hardcoded to the latency of 64-128ns.		001b	R0
5	ETFS	**Extended Tag Field Supported:** This field indicates the maximum supported size of the Tag field as a Requester. When Clear indicates 5-bit Tag field is supported. This limits the EP to maximum of 32 outstanding requests.		0b	R0
4 :3	PFS	**Phantom Functions Supported:** This field indicates the support for use of unclaimed function numbers to extend the number of outstanding transactions allowed. EP does not use this capability.		00b	RO
2 :0	MPS	**Max_Payload_Size Supported:** This field indicates the maximum payload size that EP can support for TLPs. This value is set to indicate 256B. The defined encodings are: • 000b = 128B max payload size • 001b = 256B max payload size (Max supported) • 010b - 111b = Reserved		001b	RO

20.2.5.5 PPDCNTL—PF PCI Express Device Control Register

Table 20-38. PPDCNTL—PF PCI Express Device Control Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
Size: 16 bit	**Default:** 2810h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	STARTFLR	**Initiate FLR** - Used to initiate FLR transition. A write of 1 initiates FLR transition. Since hardware must not respond to any cycles till FLR completion, the value read by software from this bit is 0.		0b	RW
14 :12	MRS	**Max_Read_Request_Size:** This field sets the maximum Read Request size for the EP as a Requester. The EP is capable for generating up to 1kB read requests. However requests generated by the EP will be limited by the programmed value in this field. Defined encodings for this field are: 000b = 128B max read request size 001b = 256B max read request size 010b = 512B max read request size (Default) 011b = 1024B max read request size 100b - 111b = Reserved		010b	RW
11	ENS	**Enable No Snoop (NS):** If this bit is set to 1, EP is permitted to set the No Snoop bit in the Requester Attributes of transactions it initiates. When clear all transactions will have the No Snoop bit clear. Setting this bit will not cause the EP to set the No Snoop attribute on every memory requests that it initiates.		1b	RW
10	APME	**Auxiliary (AUX) Power PM Enable:** This bit when set enables a device to draw AUX power independent of PME AUX power. Does not apply to EP		0b	R0
9	PFE	**Phantom Functions Enable:** When set, this bit enables a device to use unclaimed functions as Phantom Functions. Does not apply to EP.		0b	RO
8	ETFE	**Extended Tag Field Enable:** When set, this bit enables a device to use an 8-bit Tag field as a requester. Does not to EP.		0b	RO
7 :5	MPS	**Max_Payload_Size (MPS):** This field sets maximum TLP payload for EP. As a Receiver, the EP must handle TLPs as large as the set value; as a Transmitter, the EP must not generate TLPs exceeding the set value. The EP is capable of generating up to 256B MPS. However requests generated by the EP will be limited by the programmed value in this field. It is expected that the root complexes that the EP will be paired up with will be limited to 128B and 256B MPS. 000b = 128B (Default) 001b = 256B Others values not supported by EP		000b	RW
4	ERO	**Enable Relaxed Ordering:** If this bit is set, EP is permitted to set the Relaxed Ordering bit in the Attributes field of transactions it initiates. Setting this bit does not cause the EP to set the RO on every transaction it issues		1b	RW

Table 20-38. PPDCNTL—PF PCI Express Device Control Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
Size: 16 bit	**Default:** 2810h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	URRO	**Unsupported Request Reporting Enable**: This bit, in conjunction with other bits, controls the signaling of Unsupported Requests by sending Error Messages to the root port. When clear it disables sending of error messages.		0b	RW
2	FERE	**Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_FATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
1	NERE	**Non-Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_NONEATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
0	CERE	**Correctable Error Reporting Enable:** This bit, in conjunction with other bits, controls sending ERR_COR Messages to the root port. When clear disables sending of error messages.		0h	RW

20.2.5.6 PPDSTAT—PF PCI Express Device Status Register

Table 20-39. PPDSTAT—PF PCI Express Device Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :6	Reserved			0h	RO
5	TP	**Transactions Pending:** This bit when set indicates that EP has issued Non-Posted Requests which have not been completed either with a completion packet or completion timeout mechanism.		0h	RO
4	APD	**AUX Power Detected:** Devices that require AUX power report this bit as set if AUX power is detected by the device. ***Note:*** Does not apply to this EP function.		0h	RO

Table 20-39. PPDSTAT—PF PCI Express Device Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	URD	**Unsupported Request Detected:** This bit indicates that EP received an Unsupported Request. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C
2	FED	**Fatal Error Detected:** This bit indicates status of Fatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C
1	NED	**Non-Fatal Error Detected:** This bit indicates status of Nonfatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C
0	CED	**Correctable Error Detected:** This bit indicates status of correctable errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C

20.2.5.7 PLCAPR—PF Link Capabilities Register

Table 20-40. PLCAPR—PF Link Capabilities Register (Sheet 1 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 80h **Offset End:** 83h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 80h **Offset End:** 83h	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PORTNUM	**Port Number**: Assigned by EP after link training phase.		0h	RWOS
23 :22	Reserved	Reserved		00b	RO
21 :21	LBN	**Link Bandwidth Notification** Capability: ***Note:*** Does not apply to PCIe Endpoints/EP.		0b	RO
20 :18	Reserved	Reserved ***Note:*** Does not apply to EP		000b	RO

Table 20-40. PLCAPR—PF Link Capabilities Register (Sheet 2 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 80h **Offset End:** 83h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 80h **Offset End:** 83h	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17 :15	L1EL	**L1 Exit Latency**- Indicates the exit latency from L1 to L0 state. 000b - Less than 1 ìs 001b - 1 ìs - 2 ìs 010b - 2 ìs - 4 ìs 011b - 4 ìs - 8 ìs 100b - 8 ìs - 16 ìs 101b - 16 ìs - 32 ìs 110b - 32 ìs - 64 ìs 111b - L1 transition not supported ***Note:*** EP does not support ASPM L1 transition		111b	RO
14 :12	L0EL	**L0s Exit Latency**- Indicates the exit latency from L0s to L0 state. 000b - Less than 64ns 001b - 64ns - 128ns 010b - 128ns - 256ns 011b - 256ns - 512ns 100b - 512ns - 1 ìs 101b - 1 ìs - 2 ìs 110b - 2 ìs - 4 ìs 111b - Reserved		011b	RO
11 :10	ASLPM	**Active State Link PM Support** - Indicates the level of active state power management supported in EP. Defined encodings are: 00b - Reserved 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported		01b	RO
9 :4	LINKW	Max Link Width- Indicates the max link width. Relevant encoding: 000000b - Reserved 000001b - x1 000010b - x2 000100b - x4 001000b - x8 001100b - x12 010000b - x16 100000b - x32 EP value depends on SKU. However the max link width is x16.		010000b	RO
3 :0	MAXSPEED	Max Link Speed - Indicates Maximum supported Link Speed. Defined encodings are : 0001b - 2.5Gbs Link speed supported (Gen 1) 0010b - 5.0Gbs Link speed supported (Gen 2) EP indicates a max Link Speed of 5.0 Gbs.		0010b	RO

20.2.5.8 PLCNTLR—PF Link Control Register

Table 20-41. PLCNTLR—PF Link Control Register (Sheet 1 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 84h **Offset End:** 85h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 84h **Offset End:** 85h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved		0000b	RO
11	LBWINTE	**Link Autonomous Bandwidth Interrupt Enable** - When set to 1b this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been set. ***Note:*** Not applicable to EP.		0b	RO
10	LBWMINTE	**Link Bandwidth Management Interrupt Enable** - When set to 1b this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been set. ***Note:*** Not applicable to EP.		0b	RO
9	WD	**Hardware Autonomous Width Disable** - When set to 1b this bit disables hardware from changing the link width for reasons other than attempting to correct unreliable link operation by reducing link width. ***Note:*** EP does not support this feature.		0b	RO
8	ECLKPM	**Enable Clock Power Management** - not supported in EP		0b	RO
7	EXTSYNC	**Extended Sync**. When set to one, this bit forces the transmission of: - 4096 FTS ordered sets during the L0s state - followed by a single SKP ordered set prior to entering the L0 state, - as well as the transmission of 1024 TS1 ordered sets in the L1 state prior to entering the Recovery state. This mode gives external devices (e.g., logic analyzers) that may be monitoring Link activity time to achieve bit and symbol lock before the Link enters the L0 or Recovery state and resumes communication. ***Note:*** This control applies to all functions.		0b	RW
6	CCLKCFG	**Common Clock Configuration** - when set indicates that EP and the root port at the other end of the link are operating with a common reference clock. A value of 0 indicates that they operating with an asynchronous clock. This parameter affects the L0s Exit Latencies. After changing the value in this bit in both components on a Link, software must trigger the Link to retrain by writing a 1b to the Retrain Link bit. ***Note:*** This control applies to all functions.		0b	RW
5	RETRAIN	**Retrain Link**: A write of 1b to this bit initiates Link retraining by directing the Physical Layer LTSSM to the Recovery state. ***Note:*** Does not apply to endpoint/EP		0b	RO

Table 20-41. PLCNTLR—PF Link Control Register (Sheet 2 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 84h **Offset End:** 85h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 84h **Offset End:** 85h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
4	LINKDIS	**Link Disable**: This bit disables the Link by directing the LTSSM to the Disabled state when set to 1b. ***Note:*** Does not apply to endpoint/EP		0b	RO
3	RCB	**Read Completion Boundary (RCB)**: For PCIe Endpoints this field is set optionally by configuration software to indicate the RCB value of the Root Port upstream from the Endpoint. • 0b = 64B • 1b = 128 byte ***Note:*** Does not directly impact the EP.		0b	RW
2	Reserved	Reserved		0b	RO
1 :0	ASPMC	**Active State Link PM Control** - this field controls the active state PM supported on the link. Link PM functionality is determined by the lowest common denominator of all functions. Defined encodings are: 00b - PM Disabled 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported ***Note:*** This control applies to all functions.		00b	RW

20.2.5.9 PLSR—PF Link Status Register

Table 20-42. PLSR—PF Link Status Register

Description: Link Status Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 86h **Offset End:** 87h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 86h **Offset End:** 87h	
Size: 16 bit	**Default:** 0102h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved		0b	RO
14	LBWMS	**Link Bandwidth Management Status:** Not supported in EP. ***Note:*** This status is reflected in all functions.		0b	RO
13	DLACT	**Data Link Layer Link Active** Not supported by PCIe Endpoints/EP. ***Note:*** This status is reflected in all functions.		0b	RO
12	SCLKCFG	**Slot Clock Configuration** - When set indicates that EP uses the physical reference clock that the platform provides on the connector. This bit must be cleared if EP uses an independent clock. ***Note:*** This status is reflected in all functions.		0b	RWOS
11	LTINPROG	**Link Training** - Indicates that link training is in progress. Does not apply to PCIe Endpoints/EP.. ***Note:*** This status is reflected in all functions.		0b	RO
10	LTE	**Link Training Error** - Indicates that a link training error has occurred. Does not apply to PCIe Endpoints/EP. ***Note:*** This status is reflected in all functions.		0b	RO
9 :4	NLW	**Negotiated Link Width**: Negotiated Link Width - Indicates the negotiated width of the link. Relevant encoding for EP are: 000001b - x1 000010b - x2 000100b - x4 001000b - x8 010000b - x16 ***Note:*** This status is reflected in all functions.		010000b	RO
3 :0	NLS	**Negotiated Link Speed**: The negotiated Link Speed. Defined encodings are: 0001b - 2.5Gbs 0010b - 5.0Gbs ***Note:*** This status is reflected in all functions.		0010b	RO

20.2.5.10 PDCAPR2—PF Device Capabilities 2 Register

Table 20-43. PDCAPR2—PF Device Capabilities 2 Register

Description: Device Capabilities 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 98h **Offset End:** 9Bh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000012h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved.		0b	RV
4	CTODS	Completion Timeout Disable Supported. A value of 1b indicates support for the completion timeout disable mechanism.		1b	RO
3 :0	CTORS	Completion Timeout Ranges Supported: This field indicates support for the optional completion timeout programmability mechanism. This mechanism enables system software to modify the completion timeout value. Four time value ranges are defined: • Range A = 50 us to 10 ms • Range B = 10 ms to 250 ms • Range C = 250 ms to 4 s • Range D = 4 s to 64 s Bits are set according to the following table to show the timeout value ranges that are supported. • 0000b = Completion timeout programming not supported. PCH must implement a timeout value in the range 50 us to 50 ms. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A & B. • 0110b = Ranges B & C. • 0111b = Ranges A, B & C. • 1110b = Ranges B, C & D. • 1111b = Ranges A, B, C & D. • All other values are reserved. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms		0010b	RO

20.2.5.11 PDCNTR2—PF Device Control 2 Register

Table 20-44. PDCNTR2—PF Device Control 2 Register

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 9Ch **Offset End:** 9Dh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 9Ch **Offset End:** 9Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 5	Reserved	Reserved.		0b	RV
4	CTODIS	Completion Timeout Disable: When set to 1b, this bit disables the completion timeout mechanism. Software is permitted to set or clear this bit at any time. When set, the completion timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued. The default value for this bit is 0b.		0b	RW
3 :0	CTOV	Completion Timeout Value: In devices that support completion timeout programmability, this field enables system software to modify the completion timeout value. Encoding: • 0000b = Default range: 50 us to 50 ms. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. Values available if Range A (50 us to 10 ms) programmability range is supported: • 0001b = 50us to 100 us. • 0010b = 1 ms to 10 ms. Values available if Range B (10 ms to 250 ms) programmability range is supported: • 0101b = 16 ms to 55 ms. • 0110b = 65 ms to 210 ms. Values available if Range C (250 ms to 4 s) programmability range is supported: • 1001b = 260 ms to 900 ms. • 1010b = 1 s to 3.5 s. Values available if the Range D (4 s to 64 s) programmability range is supported: • 1101b = 4 s to 13 s. • 1110b = 17 s to 64 s. Values not defined are reserved. Software is permitted to change the value in this field at any time. For requests already pending when the completion timeout value is changed, hardware is permitted to use either the new or the old value for the outstanding requests and is permitted to base the start time for each request either when this value was changed or when each request was issued. The default value for this field is 0000b.		0000b	RW

20.2.5.12 PLCNTLR2—PF Link Control 2 Register

Table 20-45. PLCNTLR2—PF Link Control 2 Register (Sheet 1 of 2)

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** A4h **Offset End:** A5h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** A4h **Offset End:** A5h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved		0h	RV
12	CDE	Compliance De-emphasis – This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 1b -3.5 dB 0b -6 dB ***Note:*** This bit field is valid for function 0 only.	Y	0b	RWS
11	CSOS	Compliance SOS – When set to 1b, the LTSSM is required to send SOS periodically in between the (modified) compliance patterns. ***Note:*** This bit field is valid for function 0 only.	Y	0b	RWS
10	EMC	Enter Modified Compliance – When this bit is set to 1b, the device transmits modified compliance pattern if the LTSSM enters Polling.Compliance state. ***Note:*** This bit field is valid for function 0 only.	Y	0b	RWS
9 7	TMARG	Transmit Margin – This field controls the value of the non deemphasized voltage level at the Transmitter pins. Encodings: 000b - Normal operating range 001b - 800-1200 mV for full swing and 400-700 mV for half-swing. 010b - (n-1) - Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n: 200-400 mV for full-swing and 100-200 mV for half-swing. other values - reserved ***Note:*** This bit field is valid for function 0 only.	Y	0b	RWS
6	SDEM	Selectable De-emphasis This bit is not applicable and reserved for Endpoints.		0b	RO

Table 20-45. PLCNTLR2—PF Link Control 2 Register (Sheet 2 of 2)

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** A4h **Offset End:** A5h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A4h **Offset End:** A5h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
5	HWAUTOSD	Hardware Autonomous Speed Disable. When set to 1b, this bit disables hardware from changing the link speed for reasons other than attempting to correct unreliable link operation by reducing link speed. Hard wire to 0b. ***Note:*** This bit field is valid for function 0 only.		0b	RO
4	ENCOMP	Enter Compliance. Software is permitted to force a link to enter compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a link and then initiating a hot reset on the link. The default value of this field following a fundamental reset is 0b. ***Note:*** This bit field is valid for function 0 only.	Y	0b	RWS
3 :0	TLNKS	Target Link Speed. This field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a link into compliance mode. Defined encodings are: 0001b = 2.5 Gb/s Target Link Speed. 0010b = 5 Gb/s Target Link Speed. All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. The default value of this field is the highest link speed supported by PCH (as reported in the Supported Link Speeds field of the Link Capabilities register). ***Note:*** This bit field is valid for function 0 only.	Y	0010b	RWS

20.2.5.13 PLSR2—PF Link Status 2 Register

Table 20-46. PLSR2—PF Link Status 2 Register

Description: Link Status 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** A6h **Offset End:** A7h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A6h **Offset End:** A7h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 1	Reserved	Reserved		0h	RV
:0	CDEL	Current De-emphasis Level – When the Link is operating at 5 GT/s speed, this bit reflects the level of de-emphasis. it is undefined when the Link is operating at 2.5 GT/s speed Encodings: 1b -3.5 dB 0b -6 dB		0b	RO

20.2.6 MSI Capability Structure

20.2.6.1 PMSICID—PF Message Signalled Interrupt Capability ID Register

The Message Signalled Interrupt Capability record defines how the device generates PCI MSI messages. It is a PCI SIG-defined capability record and includes the MCID, MCP, MCTL, MADR, and MDATA fields of the configuration header.

Table 20-47. PMSICID—PF Message Signalled Interrupt Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** B0h **Offset End:** B0h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B0h **Offset End:** B0h	
Size: 8 bit	**Default:** 05h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCID	**Capability ID:** PCI SIG assigned capability record ID (05h, MSI capability)		05h	RO

20.2.6.2 PMSINCP—PF Message Signalled Interrupt Next Capability Pointer Register

Table 20-48. PMSINCP—PF Message Signalled Interrupt Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** B1h **Offset End:** B1h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B1h **Offset End:** B1h	
Size: 8 bit	**Default:** 60h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** MSI-X capability.		60h	RO

20.2.6.3 PMSICTL—PF Message Signalled Interrupt Control Register

Table 20-49. PMSICTL—PF Message Signalled Interrupt Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** B2h **Offset End:** B3h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 0180h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved		0h	RO
08	MC	**Per-Vector Masking Capable:** Per-vector masking capable.		1b	RO
07	C64	**64 bit Address Capable:** Hardwired to 1 to indicate the device generate 64b message addresses.		1b	RO
06 :04	MME	**Multiple Message Enable:** System software writes to this field to indicate the number of allocated messages (less than or equal to the number of requested messages in MMC). A value of 0 corresponds to one message.		000h	RW
03 :01	MMC	**Multiple Message Capable:** System software reads this field to determine the number of requested messages. Hardwired to 0 to request one message.		000h	RO
00	MSIE	MSI Enable: System software sets this bit to enable MSI signaling. A device driver is prohibited from writing this bit to mask a device's service request. If 1, the device can use an MSI to request service. If 0, the device cannot use an MSI to request service.		0h	RW

20.2.6.4 PMSILADDR—PF Message Signalled Interrupt Lower Address Register

Table 20-50. PMSILADDR—PF Message Signalled Interrupt Lower Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** B4h **Offset End:** B7h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B4h **Offset End:** B7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	**Lower Message Address:** Written by the system to indicate the lower 30-bits of the address to use for the MSI memory write transaction.		0h	RW
01 :00	Reserved	**Reserved**		00b	RV

20.2.6.5 PMSIUADDR—PF Message Signalled Interrupt Upper Address Register

Table 20-51. PMSIUADDR—PF Message Signalled Interrupt Upper Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** B8h **Offset End:** BBh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Message Address:** Written by the system to indicate the lower 62-bits of the address to use for the MSI memory write transaction. The lower two bits will always be written as 0.		0h	RW

20.2.6.6 PMSIDATA—PF Message Signalled Interrupt Data Register

Table 20-52. PMSIDATA—PF Message Signalled Interrupt Data Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** BCh **Offset End:** BDh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** BCh **Offset End:** BDh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	**Message Data:** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWORD transaction. The upper 16 bits of the transaction are written as 0.		0h	RW

20.2.6.7 PMSIMSK—PF Message Signalled Interrupt Mask Register

Table 20-53. PMSIMSK - PF Message Signalled Interrupt Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** C0h **Offset End:** C3h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RV
00 :00	MASK0	**Mask Bits:** Only on bit defined. See the interrupt section.		0b	RW

20.2.6.8 PMSIPND—PF Message Signalled Interrupt Pending Register

Table 20-54. PMSIPND—PF Message Signalled Interrupt Pending Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** C4h **Offset End:** C7h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RV
00 :00	Reserved	**Pending Bits:** Not used.		0b	RO

20.2.7 PF Advanced Error Reporting Capability Structure

20.2.7.1 PPCIEAERCAPID—PF PCI Express AER Capability ID Register

The PCI Express Capability List register enumerates the PCI Express AER Capability structure in the PCI 3.0 configuration space capability list.

Note: Depending on the ACC_VF_ENABLE strap the AER Capability can be the last capability in the list.

Table 20-55. PPCIEAERCAPID—PF PCI Express AER Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 100h **Offset End:** 103h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 100h **Offset End:** 103h	
Size: 32 bit	**Default:** 13810001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	PCIEAERNCP	Next PCI Express Extended Capability Pointer: The Next Capability Pointer default value will be dependent on the ACC_VF_ENABLE strap. 1 - 138h (Default) 0 - 000h (AER is the last extended capability)		138h	RO
19 :16	PCIEAERCVN	Advanced Error Capability Version Number: PCI Express Advanced Error Reporting Extended Capability Version Number.		1h	RO
15 :00	PCIEAERCID	Advanced Error Capability ID: PCI Express Extended Capability ID indicating Advanced Error Reporting Capability.		1h	RO

20.2.7.2 PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register

Table 20-56. PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 104h **Offset End:** 107h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		00b	RV
20	UR	**Unsupported Request Error Status:** As a receiver, Set whenever an unsupported request is detected. The Header is logged.	Y	0b	RW1CS
19	ECRCC	**ECRC Check:** As a receiver, set when ECRC check fails. The Header is logged. Not supported in EP	Y	0b	RW1CS
18	MTLP	**Malformed TLP**: As a receiver, set whenever a malformed TLP is detected. The Header is logged.	Y	0b	RW1CS
17	RO	**Receiver Overflow:** Set if PCI Express receive buffers overflow.	Y	0b	RW1CS
16	EC	**Unexpected Completion:** As a receiver, set whenever a completion is received that does not match the EP requestor ID or outstanding Tag. The Header is logged.	Y	0b	RW1CS
15	CA	**Completer Abort:** As a completer, set whenever an internal agent signals a data abort. The header is logged.	Y	0b	RW1CS
14	CT	**Completion Timeout**: As a requester, set whenever an outbound Non Posted Request does not receive a completion within 16-32ms.	Y	0b	RW1CS
13	FCPES	**Flow Control Protocol Error Status:** Set whenever a flow control protocol error is detected. Not supported.	Y	0b	RW1CS
12	PTLPR	**Poisoned TLP Received:** As a receiver, set whenever a poisoned TLP is received from PCI Express. The header is logged. Internal queue errors are not covered by this bit, they are logged by the Configuration target of the transaction.	Y	0h	RW1CS
11 :6	Reserved	Reserved		0000b	RV
5	SDES	**Surprise Down Error:** Not supported.	Y	0b	RW1CS
4	DLPE	**Data Link Protocol Error:** Set whenever a data link protocol error is detected.	Y	0b	RW1CS
03 :00	Reserved	Reserved		0h	RV

20.2.7.3 PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register

Table 20-57. PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 108h **Offset End:** 10Bh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		0h	RV
20	UR	**Unsupported Request Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
19	ECRCC	**ECRC Check Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
18	MTLP	**Malformed TLP Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
17	RO	**Receiver Overflow Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
16	EC	**Unexpected Completion Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
15	CA	**Completer Abort Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
14	CT	**Completion Time Out Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
13	FCPES	**Flow Control Protocol Error Mask**: When '1' error reporting is masked. Not supported.	Y	0b	RWS
12	PTLPR	**Poisoned TLP Received Error Mask**: When '1' error reporting is masked.	Y	0h	RWS
11 :6	Reserved	Reserved		0h	RV
5	SDES	**Surprise Down Error:** Not supported.	Y	0b	RWS
4	DLPE	**Data Link Protocol Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
03 :01	Reserved	Reserved		0h	RV
00	Reserved	Undefined: Reserved		0b	RV

20.2.7.4 PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register

The Uncorrectable Error Severity register controls whether an individual uncorrectable error is reported as a non-fatal or fatal error. An error is reported as fatal when the corresponding error bit in the severity register is set. If the bit is cleared, the corresponding error is considered non-fatal.

Table 20-58. PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 10Ch **Offset End:** 10Fh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00062030h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		0h	RV
20	UR	**Unsupported Request Error Status Severity(URESS):**	Y	0b	RWS
19	ECRCC	**ECRC Check Severity:** Not supported in EP	Y	0b	RWS
18	MTLP	**Malformed TLP Severity:**	Y	1b	RWS
17	RO	**Receiver Overflow Severity:**	Y	1b	RWS
16	EC	**Unexpected Completion Severity:**	Y	0b	RWS
15	CA	**Completer Abort Severity:**	Y	0b	RWS
14	CT	**Completion Time Out Severity:**	Y	0b	RWS
13	FCPES	**Flow Control Protocol Error Severity:** Not supported.	Y	1b	RWS
12	PTLPR	**Poisoned TLP Received Severity:**	Y	0h	RWS
11 :6	Reserved	Reserved		0h	RV
5	SDES	**Surprise Down Error Severity:** Not supported.	Y	1b	RWS
4	DLPE	**Data Link Protocol Error Severity:**	Y	1b	RWS
03 :01	Reserved	Reserved		0h	RV
00	Reserved	Undefined: Reserved		0b	RV

20.2.7.5 PPAERCS—PF PCI Express AER Correctable Error Register

Table 20-59. PPAERCS—PF PCI Express AER Correctable Error Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:0:0		**Offset Start:** 110h **Offset End:** 113h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		0h	RV
13	ANFES	**Advisory Non-Fatal Error Status**	Y	0b	RW1CS
12	RTTS	**Replay Timer Timeout Status**: Set whenever a replay timer timeout occurs.	Y	0b	RW1CS
11 :09	Reserved	**Reserved**		0b	RV
8	RNRS	**REPLAY NUM Rollover Status**: Set whenever the replay number rolls over from 11 to 00.	Y	0h	RW1CS
7	BDLLPS	**Bad DLLP Status**: Sets this bit on CRC errors on DLLP.	Y	0b	RW1CS
6	DLPE	**Bad TLP Status**: Sets this bit on CRC errors or sequence number out of range on TLP.	Y	0b	RW1CS
05 :01	Reserved	Reserved		0h	RV
00	RES	**Receiver Error Status**: Set whenever the physical layer detects a receiver error.	Y	0b	RW1CS

20.2.7.6 PPAERCM—PF PCI Express AER Correctable Error Mask Register

Table 20-60. PPAERCM—PF PCI Express AER Correctable Error Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 114h **Offset End:** 117h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 2000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		0h	RV
13	ANFES	**Advisory Non-Fatal Error Mask**: this bit is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.	Y	1b	RWS
12	RTTS	**Replay Timer Timeout Mask:**	Y	0b	RWS
11 :09	Reserved	**Reserved**		0h	RV
8	RNRS	**REPLAY NUM Rollover Mask:**	Y	0b	RWS
7	BDLLPS	**Bad DLLP Mask:**	Y	0b	RWS
6	DLPE	**Bad TLP Mask:**	Y	0b	RWS
05 :01	Reserved	Reserved		0h	RV
00	RES	**Receiver Error Mask:**	Y	0b	RWS

20.2.7.7 PPAERCTLCAP—PF PCI Express AER Control and Capability Register

Table 20-61. PPAERCTLCAP—PF PCI Express AER Control and Capability Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 118h **Offset End:** 11Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved		0h	RV
8	ECRCCE	**ECRC Check Enable**: When set enables ECRC checking. ECRC not supported	Y	0b	RWS
7	ECRCCC	**ECRC Check Capable**: Indicates EP is capable of checking ECRC.		0b	RO
6	ECRCGE	**ECRC Generation Enable**: When set enables ECRC generation. ECRC is not supported	Y	0b	RWS
5	ECRCGC	**ECRC Generation Capable**: Indicates the EP is not capable of generating ECRC.		0b	RO
04 :00	TFEP	**The First Error Pointer:** Identifies the bit position of the first error reported in the Section 20-56, "PPAERUCS—PF PCI Express AER Uncorrectable Error Status Register" This register will not update until all bits in the ERRUNC STS register are cleared.	Y	00000b	ROS

20.2.7.8 PPAERHDRLOG0—PF PCI Express AER Header Log 0 Register

Table 20-62. PPAERHDRLOG0—PF PCI Express AER Header Log 0 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 11Ch **Offset End:** 11Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 11Ch **Offset End:** 11Fh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW0	1st DWord of the Header for the PCI Express packet in error (HDRLOGDW0): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.2.7.9 PPAERHDRLOG1—PF PCI Express AER Header Log 1 Register

Table 20-63. PPAERHDRLOG1—PF PCI Express AER Header Log 1 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 120h **Offset End:** 123h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 120h **Offset End:** 123h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW1	Second DWord of the Header for the PCI Express packet in error (HDRLOGDW1): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.2.7.10 PPAERHDRLOG2—PF PCI Express AER Header Log 2 Register

Table 20-64. PPAERHDRLOG2—PF PCI Express AER Header Log 2 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 124h **Offset End:** 127h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 124h **Offset End:** 127h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW2	Third DWord of the Header for the PCI Express packet in error (HDRLOGDW2): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.2.7.11 PPAERHDRLOG3—PF PCI Express AER Header Log 3 Register

Table 20-65. PPAERHDRLOG3—PF PCI Express AER Header Log 3 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 128h **Offset End:** 12Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 128h **Offset End:** 12Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRDWLOG3	4th DWord of the Header for the PCI Express packet in error (HDRDWLOG3): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.2.8 PF Alternative Routing-ID Extended Capability Structure

This section describes the PCI Express Extended Configuration Space registers that make up the Alternative Routing ID Extended Capability Structure.

Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. See the PCI Express* Base Specification 2.0 for the full register descriptions and additional information regarding their operation.

20.2.8.1 PARIDHDR—PF Alternative Routing ID Capability Header

This register contains information associated with the Alternative Routing ID capability. This is compliant with the *PCI-SIG ECN: Alternatifve Routing-ID Interpretation (ARI)*, Updated June 4, 2007.

20.2.8.2 PFARICAP—PF ARI Capabilities Register

Table 20-66. PARIDHDR—PF Alternative Routing ID Capability Header

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 138h **Offset End:** 13Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 138h **Offset End:** 13Bh	
Size: 32 bit	**Default:** 1401000Eh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ARINCO	Next Capability Offset - This field contains 140h which points to the next item in the extended capabilities list, the SR-IOV extended capability.		140h	RO
19 :16	ARICV	Capability Version - This is set to 1h for the most current version of the specification.		1h	RO
15 :0	ARICV	PCI Express Extended Capability ID - The PCI SIG has assigned 000Eh to the ARI extended capability.		000EH	RO

This register contains information associated with the Alternative Routing ID capability.

Table 20-67. PFARICAP—PF ARI Capabilities Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 13Ch **Offset End:** 13Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0100h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. The Next Function Number is set based on fuse and EEPROM bits. However, some of the values listed below will never be observed as some fuse and EEPROM bits' combinations are not valid scenarios. 0h - If only Function 0 is enabled. 1h - If Function 1 is enabled. 2h - If Function 1 is disabled, but Function 2 enabled. 3h - If Functions 1-2 are disabled, but Function 3 is enabled. 4h - If Function 1-3 are disabled, but Function 4 is enabled.		1h	RO
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: Hardwired to Zero .		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: Hardwired to Zero		0b	RO

20.2.8.3 PARIDCTL—PF Alternative Routing ID Control Register

This register contains information associated with the Alternative Routing ID capability.

Table 20-68. PARIDCTL—PF Alternative Routing ID Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 13Eh **Offset End:** 13Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 13Eh **Offset End:** 13Fh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :7	Reserved	Reserved		0h	RV
6 :4	FG	**Function Group:** Hardwired to Zero.		0b	RO
3 :2	Reserved	Reserved		0b	RV
1	ACS	**ACS Functional Groups Enable**: Hardwired to Zero.		0b	RO
0	MFVC	**MFVC Functional Groups Enable**: Hardwired to Zero.		0b	RO

20.2.9 PF SR-IOV Extended Capability Structure

20.2.9.1 PSRIOVCAPID—PF SR-IOV Capability ID Register

Table 20-69. PSRIOVCAPID—PF SR IOV Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 140h **Offset End:** 141h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 140h **Offset End:** 141h	
Size: 16 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SRIOVOD	**Capability ID:** PCI SIG assigned a capability record ID for SR-IOV per the 1.0 revision		0010h	RO

20.2.9.2 PSRIOVCVNC—PF SRIOV Capability Version and Next Capability Pointer Register

Table 20-70. PSRIOVCVNC—PF SRIOV Capability Version and Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 142h **Offset End:** 143h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 142h **Offset End:** 143h	
Size: 16 bit	**Default:** 1h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :4	SRIOVNCO	Next Capability Offset - last extended capability.		00h	RO
3 :0	SRIOVCV	Capability Version - This is set to 1h for the Single Root I/O Virtualization and Sharing Specification, Revision 0.9.		1h	RO

20.2.9.3 PSRIOVCAP—PF SRIOV Capabilities Register

20.2.9.4 PSRIOVCS—PF SRIOV Control and Status Register

Table 20-71. PSRIOVCAP—PF SRIOV Capabilities Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 144h **Offset End:** 147h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 144h **Offset End:** 147h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	VFMINT	VF Migration Interrupt Message Number - Indicates the MSI vector used for migration interrupts. The value in this field has no significance if bit 0 of this capability is Clear. ***Note:*** N/A for EP		0h	RO
20 :1	Reserved	Reserved.		0h	RO
:0	VFMCAP	VF Migration Capable - If Set, the PF is Migration Capable and operating under a Migration Capable MR-PCIM. ***Note:*** EP does not support VF migration		0b	RO

20.2.9.5 PSRIOVMTOTINI—PF SRIOV Initial and Total VFs Register

Table 20-72. PSRIOVCS—PF SRIOV Control and Status Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 148h **Offset End:** 14Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 148h **Offset End:** 14Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved		0h	RO
4	VFARI	ARI Capable Hierarchy - Device may locate VFs in Function numbers 8 to 255 of the captured Bus number. EP supports ARI and will locate the VF1-VF16 in Function 8 through Function 23.		0b	RW
3	VFMSE	VF MSE - Memory Space Enable for Virtual Functions.		0b	RW
2	VFMIE	VF Migration Interrupt Enable - Enables / Disables VF Migration State Change Interrupt. ***Note:*** EP does not support VF Migration		0b	RO
1	VFME	VF Migration Enable - Enables / Disables VF Migration Support. ***Note:*** EP does not support VF Migration		0b	RO
0	VFE	VF Enable - Enables / Disables VFs. Default value is 0b		0b	RW

20.2.9.6 PSRIOVNUMVF—PF SRIOV Number of VFs Register

Table 20-73. PSRIOVMTOTINI—PF SRIOV Initial and Total VFs Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 14Ch **Offset End:** 14Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 14Ch **Offset End:** 14Fh	
Size: 32 bit	**Default:** 00100010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	TOTVF	TotalVFs indicates the maximum number of VFs that could be associated with the PF. Since EP does not support VF migration this field must be equal to the INITVF field		10h	RO
15 :0	INITVF	InitialVFs indicates the number of VFs that are initially associated with the PF. Since EP does not support VF migration this field must be equal to the TOTVF field		10h	RO

20.2.9.7 PSRIOVFVFO—PF SRIOV First VF Offset Register

Table 20-74. PSRIOVNUMVF—PF SRIOV Number of VFs Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 150h **Offset End:** 153h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 150h **Offset End:** 153h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved		00h	RO
23 :16	FUNDEP	This field describes dependencies between PFs. VF dependencies are the same as the dependencies of their associated PFs. If a PF is independent from other PFs of a Device, this field shall contain the Function Number of the PF. ***Note:*** EP implements a single PF and therefore this field is N/A to EP.		00h	RO
15 :0	NUMVF	Num VFs defines the number of VFs software has assigned to the PF. Software sets NumVFs to any value between 1 and TotalVFs (16 for DH89xxCC; 32 for DH89xxCL) as part of the process of creating VFs. NumVFs VFs shall be visible in the PCI-Express fabric after both NumVFs is set to a valid value and VF Enable is set to one. "Visible in the PCI-Express fabric" means that the VF shall respond to PCI Express transactions targeting the VF, following all other rules defined by this specification and the base specification.		0	RW

20.2.9.8 PSRIOVVFS—PF SRIOV VF Stride Register

Table 20-75. PSRIOVFVFO—PF SRIOV First VF Offset Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 154h **Offset End:** 155h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 154h **Offset End:** 155h	
Size: 16 bit	**Default:** 0008h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :0	VFOFFSET	First VF Offset is a constant and defines the Routing ID (RID) offset of the first VF that is associated with the PF that contains this Capability structure. The first VFs 16-bit RID is calculated by adding the contents of this field to the RID of the PF. ***Note:*** For the EP, the RID of the first VF is at an offset of eight from the RID of the Physical Function (PF).		0008h	RO

20.2.9.9 PSRIOVFDID—PF SRIOV VF Device ID Register

Table 20-76. PSRIOVVFS—PF SRIOV VF Stride Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 156h **Offset End:** 157h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 156h **Offset End:** 157h	
Size: 16 bit	**Default:** 0001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :0	VFSTRIDE	VF Stride is a constant and defines the Routing ID (RID) offset from one VF to the next one for all VFs associated with the PF that contains this Capability structure. The next VFs 16-bit RID is calculated by adding the contents of this field to the RID of the current VF.		1h	RO

20.2.9.10 PSRIOVPAGESIZE—PF SRIOV Supported Page Size Register

Table 20-77. PSRIOVFDID—PF SRIOV VF Device ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 158h **Offset End:** 15Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 158h **Offset End:** 15Bh	
Size: 32 bit	**Default:** 04420000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 16	VFDID151	VF Device ID - The Device ID that is presented to the OS for every VF. PCH VF DID = 0442h.		0442h	RWOS
15 :00	Reserved	Reserved		0h	RV

20.2.9.11 PSRIOVSYSPS—PF SRIOV System Page Size Register

Table 20-78. PSRIOVPAGESIZE—PF SRIOV Supported Page Size Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 15Ch **Offset End:** 15Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 15Ch **Offset End:** 15Fh	
Size: 32 bit	**Default:** 00000553h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FIRSTVFO	This field must define the supported page sizes. This PF supports a page size of 2^(n+12) if bit n is Set. For example, if bit 0 is Set, the PF supports 4k byte page sizes. PFs are required to support 4k, 8k, 64k, 256k, 1M and 4M page sizes. All other page sizes are optional. A page size describes the minimum alignment requirements for VF BAR resources as described in "System Page Size".		00000553h	RO

20.2.9.12 PSRIOVLBAR0—SRIOV Lower BAR0 Register

Table 20-79. PSRIOVSYSPS—PF SRIOV System Page Size Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0	**Offset Start:** 160h **Offset End:** 163h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 160h **Offset End:** 163h	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SYSPAGSIZ	This field must define the page size the system will use to map the VFs' memory addresses. Software must set the value of the System Page Size to one of the page sizes set in the Supported Page Sizes field. As with Supported Page Sizes, if bit n is Set in System Page Size, the VFs associated with this PF are required to support a page size of 2^(n+12). For example, if bit 1 is Set, the system is using an 8k byte page size. The results are undefined if more than one bit is set in System Page Size. The results are undefined if a bit is Set in System Page Size that is not Set in Supported Page Sizes. When System Page Size is set, the VF associated with this PF is required to align all BAR resources on a System Page Size boundary. Each VF BARn or VF BARn pair (described below) shall be aligned on a System Page Size boundary. Each VF BARn or VF BARn pair defining a non-zero address space shall be sized to consume an integer multiple of System Page Size bytes. All data structures requiring page size alignment within a VF shall be aligned on a System Page Size boundary.		00000001h	RW

The PSRIOVLBAR0 points to the Ring Controller rings. The size of this BAR is dependent on the System Page Size. However, the programmed value is dependent on the System Page Size and the Number of VFs supported. Each VF is mapped to one bundle (16 rings).

20.2.9.13 PSRIOVUBAR0—SRIOV Upper BAR0 Register

Table 20-80. PSRIOVLBAR0—SRIOV Lower BAR0 Register

Description:						
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0		**Offset Start:** 164h **Offset End:** 167h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0		**Offset Start:** 164h **Offset End:** 167h	
Size: 32 bit	**Default:** 00000004h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	VFBAR0	**Lower SRIOV Base Address 0:** These bits are used to define the actual locations of the Ring bundles when addressed from the PCI bus. *PCI Local Bus Specification*, Revision 3.0 compliant scanning of SRIOV Base Address 0, at a minimum reveals a 4 KB memory window 0 for PSRIOVBAR0. However, this memory window size will be dictated by the System Page Size as described in Section 20.2.9.11, "PSRIOVSYSPS—PF SRIOV System Page Size Register" System Page Size / Bits Above bit 12 that respond as Reserved Memory / Window Size 4-KB / None / 64-KB 8-KB / 12:12 / 128-KB 64-KB / 15:12 / 1024-KB 256-KB / 17:12 / 4096-KB 1-MB / 19:12 / 16-MB 4-MB / 21:12 / 64-MB			0	RW
11 :04	VFZERO	**Lower Bits:** Hardwired to 0 (8KB region).			0h	RO
3	VFPREF	**Prefetchable:** Hardwired to 0 to indicate that the region is not prefetchable.			0h	RO
02 :01	VFTYP	**Addressing Type:** Hardwired to indicate a 64-bit region.			10b	RO
00	VFMEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.			0h	RO

The PSRIOVUBAR0 points to the Ring Controller rings. The size of this BAR is dependent on the System Page Size. However, the programmed value is dependent on the System Page Size and the Number of VFs supported. Each VF is mapped to one bundle (16 rings).

20.2.9.14 PSRIOVLBAR1—SRIOV Lower BAR1 Register

Table 20-81. PSRIOVUBAR0—SRIOV Upper BAR0 Register

Description:						
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0		**Offset Start:** 168h **Offset End:** 16Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0		**Offset Start:** 168h **Offset End:** 16Bh	
Size: 32 bit	**Default:** 00000000h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VFBAR0	**Upper SRIOV Base Address 0:** These bits are used to define the actual locations of the Ring bundles when addressed from the PCI bus. *PCI Local Bus Specification*, Revision 3.0 compliant scanning of SRIOV Base Address 0, at a minimum reveals a 4 KB memory window 0 for PSRIOVBAR0. However, this memory window size will be dictated by the System Page Size as described in Section 20.2.9.11, "PSRIOVSYSPS—PF SRIOV System Page Size Register" System Page Size / Bits Above bit 12 that respond as Reserved Memory / Window Size 4-KB / None / 4-KB 8-KB / 12:12 / 8-KB 64-KB / 15:12 / 64-KB 256-KB / 17:12 / 256-KB 1-MB / 19:12 / 1-MB 4-MB / 21:12 / 4-MB			0h	RW

20.2.9.15 PSRIOVUBAR1—SRIOV Upper BAR1 Register

Table 20-82. PSRIOVLBAR1—SRIOV Lower BAR1 Register

Description:						
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0		**Offset Start:** 16Ch **Offset End:** 16Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0		**Offset Start:** 16Ch **Offset End:** 16Fh	
Size: 32 bit	**Default:** 00000004h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	VFBAR1	**Lower SRIOV Base Address 1:** These bits are used to define the actual locations of the PF2VF[1-16] - PF to VF[1:16] Doorbell and Interrupt Register when addressed from the PCI bus. *PCI Local Bus Specification*, Revision 3.0 compliant scanning of SRIOV Base Address 1, at a minimum reveals a 4 KB memory window 1 for PSRIOVBAR1. However, this memory window size will be dictated by the System Page Size as described in Section 20.2.9.11, "PSRIOVSYSPS—PF SRIOV System Page Size Register" System Page Size / Bits Above bit 12 that respond as Reserved Memory / Window Size 4-KB / None / 4-KB 8-KB / 12:12 / 8-KB 64-KB / 15:12 / 64-KB 256-KB / 17:12 / 256-KB 1-MB / 19:12 / 1-MB 4-MB / 21:12 / 4-MB			0h	RW
11 :04	VFZERO	**Lower Bits:** Hardwired to 0 (4KB region)			0h	RO
3	VFPREF	**Prefetchable:** Hardwired to 0 to indicate that the region is not prefetchable.			0h	RO
02 :01	VFTYP	**Addressing Type:** Hardwired to indicate a 64-bit region.			10b	RO
00	VFMEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.			0h	RO

20.2.9.16 PSRIOVVFMA—PF SRIOV VF Migration Array Register

Table 20-83. PSRIOVUBAR1—SRIOV Upper BAR1 Register

Description:						
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0		**Offset Start:** 170h **Offset End:** 173h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0		**Offset Start:** 170h **Offset End:** 173h	
Size: 32 bit	**Default:** 00000000h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VFBAR1	**Upper SRIOV Base Address 1:** These bits are used to define the actual locations of the PF2VF[1-16] - PF to VF[1:16] Doorbell and Interrupt Register when addressed from the PCI bus. *PCI Local Bus Specification*, Revision 3.0 compliant scanning of SRIOV Base Address 1, at a minimum reveals a 4 KB memory window 1 for PSRIOVBAR1. However, this memory window size will be dictated by the System Page Size as described in Section 20.2.9.11, "PSRIOVSYSPS—PF SRIOV System Page Size Register" System Page Size / Bits Above bit 12 that respond as Reserved Memory / Window Size 4-KB / None / 4-KB 8-KB / 12:12 / 8-KB 64-KB / 15:12 / 64-KB 256-KB / 17:12 / 256-KB 1-MB / 19:12 / 1-MB 4-MB / 21:12 / 4-MB			0h	RW

Table 20-84. PSRIOVVFMA—PF SRIOV VF Migration Array Register

Description:						
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:0:0		**Offset Start:** 17Ch **Offset End:** 17Fh	
Size: 32 bit	**Default:** 00000000h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VFMA	N/A ***Note:*** Not supported in EP			00000000h	RO

20.3 EP VF PCI Configuration Space

Highlighted regions of the configurations space are RO for the VF[6:0] configuration space and have the same value as the PF.

20.3.1 PCI Standard Header Registers

This section describes the PCI Configuration Space registers that make up the standard Type 0 header. Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. See the PCI Express* Base Specification 2.0 and PCI Local Bus Specification for the full register descriptions and additional information regarding their operation.

20.3.1.1 VVID[0:15]—VF Vendor Identification Register

The VID Register contains the vendor identification number. This 16-bit register combined with the Device Identification Register uniquely identifies any PCI device. Writes to this register have no effect.

20.3.1.2 VDID[0:15]—VF Device Identification Register

Table 20-85. VVID[0:15]—VF Vendor Identification Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** FFFFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor ID: The Single Root I/O Virtualization and Sharing Specification, Revision 0.9 requires that this field return FFFFH.		FFFFh	RO

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

Table 20-86. VDID[0:15]—VF Device Identification Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** FFFFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device ID: The Single Root I/O Virtualization and Sharing Specification, Revision 0.9 requires that this field return FFFFH. Software should return the VF Device ID value from the associated PF as the Device ID for the VF.		FFFFh	RO

20.3.1.3 VPCICMD[0:15]—VF Device Command Register

Table 20-87. VPCICMD[0:15]—VF Device Command Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		0h	RV
10	INTD	Interrupt Disable: The Single Root I/O Virtualization and Sharing Specification, Revision 0.9 requires that this field is hardwired to 0b for all VFs. This bit does not apply to VFs.		0h	RO
9	FBTB	Fast Back to Back Enable: Does not apply to PCI Express. Hard-wired to 0		0h	RO
8	SER	**SERR# Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 0b for all VFs. In addition the functionality associated with the setting of this bit in the Section 20.2.2.3, "PPCICMD—PF Device Command Register" will apply to all VFs.		0h	RO
7	Reserved	Address/Data Stepping Control: Does not apply to PCI Express. Hard-wired to 0.		0h	RO
6	PER	**Parity Error Response**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 0b for all VFs. In addition the functionality associated with the setting of this bit in the Section 20.2.2.3, "PPCICMD—PF Device Command Register" will apply to all VFs.		0h	RO
5	VPS	**VGA Palette Snoop Enable:** Does not apply to PCI Express. Hard-wired to 0.		0h	RO
4	MWE	**Memory Write and Invalidate Enable**: Does not apply to PCI Express. Hard-wired to 0.		0h	RO
3	SS	**Special Cycle Enable:** Does not apply to PCI Express. Hard-wired to 0.		0h	RO
2	BM	**Bus Master Enable:** When cleared, the EP is prevented from issuing any memory or I/O read/write requests. Requests other than memory or I/O requests are not controlled by this bit. The EP will initiate a completion transaction regardless of the setting.		0h	RW
1	MEM	**Memory Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 0b for all VFs.		0h	RO
0	IO	**I/O Space Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 0b for all VFs.		0h	RO

20.3.1.4 VPCISTS[0:15]—VF Device Status Register

Table 20-88. VPCISTS[0:15]—VF PCI Device Status Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	**Detected Parity Error:** set when the EP receives a poisoned TLP regardless of the state of the Parity Error Response in the VPCICMD register.		0h	RW1C
14	SSE	**SERR# Asserted:** set when the EP sends an ERR_FATAL or ERR_NONFATAL message, and the SERR Enable bit in the VPCICMD register is '1'.		0h	RW1C
13	RMA	**Received Master Abort:** set when the EP receives a completion with Unsupported Request Completion Status.		0h	RW1C
12	RTA	**Received Target Abort:** set when the EP receives a completion with Completer Abort Completion Status.		0h	RW1C
11	STA	**Signaled Target Abort:** set when the EP completes a Request using Completer Abort Completion Status		0h	RW1C
10 :09	DST	DEVSEL# **Timing:** Does not apply to PCI Express. Hard-wired to 0.		00b	RO
8	MDPE	**Master Data Parity Error:** This bit is set by the EP if its Parity Error Enable bit is set and either of the following two conditions occurs: This bit is set under the following conditions. - EP receives a Poisoned Completion for an Outbound Read Request - EP transmits a Poisoned TLP for an Outbound Write Request. If the Parity Error Response bit is cleared in theSection 20.2.2.3, "PPCICMD—PF Device Command Register", this bit is never set.		0h	RW1C
7	FB2B	**Fast Back-to-Back**: ***Note:*** Does not apply to PCI Express.Hard-wired to 0.		0h	RO
6	Reserved	Reserved		0h	RV
5	MC66	**66 MHz Capable (C66):** ***Note:*** Does not apply to PCI Express. Hard-wired to 0		0h	RO
4	CL	**Capabilities List:** All PCI Express devices are required to implement the PCI Express capability structure. ***Note:*** Hard-wired to 1.		1h	RO
3	IS	**Interrupt Status:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 0b for all VFs. ***Note:*** This bit does not apply to VFs.		0h	RO
02 : 00	Reserved	Reserved		0h	RV

20.3.1.4.1 VRID[0:15]—Revision ID Register

The value of this register indicates the chip stepping. It is hardwired on chip and reflects the latest revision.

Table 20-89. VRID[0:15]—VF Revision ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :0	RID	**EP Revision:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field should be viewed as a Vendor Defined Extension to the Device ID. This may be different than the PF's Revision ID, but the same value must be reported by all VFs. See the PF RID for a description.		10h	RO

20.3.1.5 VCC[0:15]—VF Class Code Register

Table 20-90. VCC[0:15]—VF Class Code Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 09h **Offset End:** 0Bh	
Size: 24 bit	**Default:** 0B4000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :0	CC	**Class Code:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field return the same value as the Section 20.2.2.5, "PCC—PF Class Code Register" for all VFs.		0B4000h	RWOS

20.3.1.6 VHDR[0:15]—VF Header Type Register

Table 20-91. VHDR[0:15]—VF Header Type Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HDR	**Header Type:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field must be set to 00H for VFs.		00h	RO

20.3.1.7 VSVID[0:15]—VF Subsystem Vendor ID Register

Table 20-92. VSVID[0:15]—VF Subsystem Vendor ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	**Subsystem Vendor ID:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that when read, this read only register must return the same value as the Section 20.2.2.13, "PSVID—PF Subsystem Vendor ID Register" for all VFs.		8086h	RO

20.3.1.8 VSID[0:15]—VF Subsystem ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

20.3.1.9 VCP[0:15]—VF Capabilities Pointer Register

Table 20-93. VSID[0:15]—VF Subsystem ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	**Subsystem ID:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this read only register return the same value for all VFs. This may be a different value than that contained in the PF (e.g.,Section 20.2.2.14, "PSID—PF Subsystem ID Register").		0000h	RWOS

The Capabilities Pointer Register provides the offset in configuration space to the location where the first set of capabilities registers is located.

Table 20-94. VCP[0:15]—VF Capabilities Pointer Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 90h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	**Capability List Pointer:** This field provides an offset into the EP's configuration space pointing to the first item in the EP capability list which in the EP VF is the MSI extended capabilities header.		90h	RO

20.3.1.10 VIRQL[0:15]—VF Interrupt Line Register

Table 20-95. VIRQL[0:15]—VF Interrupt Line Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQL	**Interrupt Assigned:** This field does not apply to VFs and is hardwired to Zero.		0h	RO

20.3.1.11 VIRQP[0:15]—VF Interrupt Pin Register

Table 20-96. VIRQP[0:15]—VF Interrupt Pin Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQP	**Interrupt Assigned:** ***Note:*** This field does not apply to VFs and is hardwired to Zero.		0h	RO

20.3.2 VF PCI Express Capability Structure

20.3.2.1 VPCID[0:15]—VF PCI Express Capability ID Register

The PCI Express Capability List register enumerates the PCI Express Capability structure in the PCI 3.0 configuration space capability list.

Table 20-97. VPCID[0:15]—VF PCI Express Capability ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 50h **Offset End:** 50h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIECID	**Cap Id:** This field identifies this item in the linked list of Extended Capability Headers as being the PCI Express capability registers.		10h	RO

20.3.2.2 VPCP[0:15]—VF PCI Express Next Capability Pointer Register

Table 20-98. VPCP[0:15]—VF PCI Express Next Capability Pointer Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 51h **Offset End:** 51h	
Size: 8 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Last capability.		0h	RO

20.3.2.3 VPCR[0:15]—VF PCI Express Capabilities Register

Table 20-99. VPCR[0:15]—VF PCI Express Capabilities Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** 002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved		00b	RO
13 :9	IMN	**Interrupt Message Number**: This only applies to Root Complex and Switch devices. This register is hardcoded to 0.		00000b	RO
8	SI	**Slot Implemented:** Indicates that the PCI Express Link associated with this port is connected to a slot. Only valid for root complex and switch downstream ports. Hard-wired to 0		0b	RO
7 :4	DPT	**Device/Port Type:** Indicates the type of PCI Express logical device. 0000b - PCI Express Endpoint device		0000b	RO
3 :0	CV	**Capability Version:** Indicates PCI-SIG defined PCI Express capability structure version number EP supports version 2h.		0010b	RO

20.3.2.4 VPDCAP[0:15]—VF PCI Express Device Capabilities Register

Table 20-100. VPDCAP[0:15]—VF PCI Express Device Capabilities Register (Sheet 1 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 54h **Offset End:** 54h	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved	Reserved		0h	RO
28	FLR	**FLR Cap:** Function Level Reset Capability is required for all VFs and PFs according to the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9.		1b	RO
27 :26	CSPS	**Captured Slot Power Limit Scale:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field is undefined for all VFs.		00b	RO
25 :18	CSPV	**Captured Slot Power Limit Value**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field is undefined for all VFs.		0h	RO
17 :16	Reserved	Reserved		0h	RV
15	RBEP	**Role-Based Error Reporting**: this bit is set to indicate that this device implements the Role Base Error Reporting defined in *PCI Express Base Specification*, Revision 2.0.		1b	RO
14 :12	Reserved	**Reserved:** Undefined - Treated as Reserved		000b	RV

Table 20-100.VPDCAP[0:15]—VF PCI Express Device Capabilities Register (Sheet 2 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 54h **Offset End:** 54h	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11 :9	EL1L	**Endpoint L1 Acceptable Latency:** EP does not support L1 active state power management.		000b	RO
8 :6	EL0L	**Endpoint L0 Acceptable Latency:** Total acceptable latency that the EP can withstand due to a transition from L0s to L0 state.		001b	RO
5	ETFS	**Extended Tag Field Supported:** Indicates the maximum supported size of the Tag field as a Requester. EP does not generate 8-bit Tags but supports 8-bit Tags as a completer.		0b	RO
4 :3	PFS	**Phantom Functions Supported:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to 00b.		00b	RO
2 :0	MPS	**Max_Payload_Size Supported:** This field indicates the maximum payload size that EP can support for TLPs. This value is set to indicate 256B. The defined encodings are: • 000b = 128B max payload size • 001b = 256 bytes max payload size (Max supported) • 010b - 111b = Reserved		001b	RO

20.3.2.5 VPDC[0:15]—VF PCI Express Device Control Register

Table 20-101.VPDC[0:15]—VF PCI Express Device Control Register (Sheet 1 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 58h **Offset End:** 59h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	STARTFLR	**Initiate Function Level Reset:** A write of 1b to this bit initiates Function Level Reset to the EP. The value is always read as 0b.		0b	RW
14 :12	MRS	**Max Read Request Size:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		000b	RO
11	ENS	**Enable No Snoop:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
10	APME	**Aux Power PM Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	R0
9	PFE	**Phantom Functions Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
8	ETFE	**Extended Tag Field Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO

Table 20-101.VPDC[0:15]—VF PCI Express Device Control Register (Sheet 2 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 58h **Offset End:** 59h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
7 :5	MPS	**Max Payload Size:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		000b	RO
4	ERO	**Enable Relaxed Ordering:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
3	URRO	**Unsupported Request Reporting Enable (URRE):** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL—PF PCI Express Device Control Register", and will apply to all VFs.		0b	RO
2	FERE	**Fatal Error Reporting Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL—PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO
1	NERE	**Non-Fatal Error Reporting Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL—PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO
0	CERE	**Correctable Error Reporting Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL—PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO

20.3.2.6 VPDS[0:15]—VF PCI Express Device Status Register

Table 20-102.VPDS[0:15]—VF PCI Express Device Status Register (Sheet 1 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :6	Reserved			0h	RV
5	TP	**Transactions Pending:** This bit when set indicates that a device has issued Non-Posted Requests which have not been completed. A device reports this bit cleared only when all Completions for any outstanding Non-Posted Requests have been received.		0h	RW1C
4	APD	**AUX Power Detected:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0h	RO

Table 20-102.VPDS[0:15]—VF PCI Express Device Status Register (Sheet 2 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	URD	**Unsupported Request Detected:** This bit indicates that the device received an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW1C
2	FED	**Fatal Error Detected:** This bit indicates status of Fatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW1C
1	NED	**Non-Fatal Error Detected:** This bit indicates status of Nonfatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW1C
0	CED	**Correctable Error Detected:** This bit indicates status of correctable errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW1C

20.3.2.7 VLCR[0:15]—VF Link Capabilities Register

Table 20-103.VLCR[0:15]—VF Link Capabilities Register (Sheet 1 of 2)

Description: Link Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ch **Offset End:** 5Fh	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PORTNUM	**Port Number**: Assigned by EP after link training phase.		0h	RO
23 :22	Reserved	Reserved		00b	RO
21 :21	Reserved	Reserved		0b	RO
20 :18	Reserved	Reserved		000b	RO

Table 20-103.VLCR[0:15]—VF Link Capabilities Register (Sheet 2 of 2)

Description: Link Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ch **Offset End:** 5Fh	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17 :15	L1EL	**L1 Exit Latency**- Indicates the exit latency from L1 to L0 state. EP does not support L1 transition 000b - Less than 1 ìs 001b - 1 ìs - 2 ìs 010b - 2 ìs - 4 ìs 011b - 4 ìs - 8 ìs 100b - 8 ìs - 16 ìs 101b - 16 ìs - 32 ìs 110b - 32 ìs - 64 ìs 111b - L1 transition not supported		111b	RO
14 :12	L0EL	**L0s Exit Latency**- Indicates the exit latency from L0s to L0 state. 000b - Less than 64ns 001b - 64ns - 128ns 010b - 128ns - 256ns 011b - 256ns - 512ns 100b - 512ns - 1 ìs 101b - 1 ìs - 2 ìs 110b - 2 ìs - 4 ìs 111b - Reserved		011b	RO
11 :10	ASLPM	**Active State Link PM Support** - Indicates the level of active state power management supported in EP. Defined encodings are: 00b - Reserved 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported		01b	RO
9 :4	LINKW	**Max Link Width**- Indicates the max link width. Relevant encoding: 000000b - Reserved 000001b - x1 000010b - x2 000100b - x4 001000b - x8 001100b - x12 010000b - x16 100000b - x32 EP value depends on SKU. However the max link width is x16.		010000b	RO
3 :0	MAXSPEED	**Max Link Speed** - Indicates Maximum supported Link Speed. Defined encodings are : 0001b - 2.5Gbs Link speed supported (Gen 1) 0010b - 5.0Gbs Link speed supported (Gen 2) EP indicates a max Link Speed of 5.0 Gbs.		0010b	RO

20.3.2.8 VLCNTLR[0:15]—VF Link Control Register

Table 20-104.VLCNTLR[0:15]—VF Link Control Register

Description: Link Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 60h **Offset End:** 61h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved		0000b	RV
11	Reserved	Reserved		0b	RO
10	Reserved	Reserved		0b	RO
9	Reserved	Reserved: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
8	ECLKPM	The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
7	EXTSYNC	**Extended Synch**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
6	CCLKCFG	**Common Clock Configuration:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.r		0b	RO
5	RETRAIN	**Retrain Link**: Not Applicable to endpoints. Hard-wired to 0		0b	RO
4	LINKDIS	**Link Disable:** Not Applicable to endpoints. Hard-wired to 0		0b	RO
3	RCB	**Read Completion Boundary (RCB) Control:**The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		0b	RO
2	Reserved	Reserved		0b	RO
1 :0	ASPMC	**Active State PM Control:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field is hardwired to all zeros.		00b	RO

20.3.2.9 VLSR[0:15]—VF Link Status Register

For the VF, all settings in this field are reserved and the PF setting in the Section 20.2.5.9, "PLSR—PF Link Status Register" applies to all of the VFs.

20.3.2.10 VDCAPR2[0:15]—VF Device Capabilities 2 Register

Table 20-105.VLSR[0:15]—VF Link Status Register

Description: Link Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 62h **Offset End:** 63h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	Reserved	Reserved		0H	RV

Table 20-106.DCAPR2[0:15]—Device Capabilities 2 Register

Description: Device Capabilities 2 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Device:Function:** M:8 + Index1	**Offset Start:** 74h **Offset End:** 77h	
Size: 32 bit	**Default:** 00000012h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved.		0b	RV
4	CTODS	Completion Timeout Disable Supported. A value of 1b indicates support for the completion timeout disable mechanism. The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field value must be identical as the PF value.		1b	RO
3 :0	CTORS	Completion Timeout Ranges Supported: This field indicates support for the optional completion timeout programmability mechanism. This mechanism enables system software to modify the completion timeout value. Four time value ranges are defined: • Range A = 50 us to 10 ms • Range B = 10 ms to 250 ms • Range C = 250 ms to 4 s • Range D = 4 s to 64 s Bits are set according to the following table to show the timeout value ranges that are supported. • 0000b = Completion timeout programming not supported. PCH must implement a timeout value in the range 50 us to 50 ms. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A & B. • 0110b = Ranges B & C. • 0111b = Ranges A, B & C. • 1110b = Ranges B, C & D. • 1111b = Ranges A, B, C & D. • All other values are reserved. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field value must be identical as the PF value.		0010b	RO

20.3.3 VF MSI Capability Structure

20.3.3.1 VMSICID[0:15]—Message Signalled Interrupt Capability ID Register

The Message Signalled Interrupt Capability record defines how the device generates PCI MSI messages. It is an 10B PCI SIG-defined capability record and includes the MCID, MCP, MCTL, MADR, and MDATA fields of the configuration header.

Table 20-107.VMSICID[0:15]—Message Signalled Interrupt Capability ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 90h **Offset End:** 90h	
Size: 8 bit	**Default:** 05h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCID	**Capability ID:** PCI SIG assigned capability record ID (05h, MSI capability)		05h	RO

20.3.3.2 VMSINCP[0:15]—Message Signalled Interrupt Next Capability Pointer Register

Table 20-108.VMSINCP[0:15]—Message Signalled Interrupt Next Capability Pointer Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 91h **Offset End:** 91h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Next Capability is PCI Express.		00h	RO

20.3.3.3 VMSICTL[0:15]—Message Signalled Interrupt Control Register

Table 20-109.VMSICTL[0:15]—Message Signalled Interrupt Control Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 92h **Offset End:** 93h	
Size: 16 bit	**Default:** 0180h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved		0h	RO
08	MC	**Per-Vector Masking Capable:** Per-vector masking capable.		1b	RO
07	C64	**64 bit Address Capable:** Hardwired to 0 to indicate the device does not generate 64b message addresses.		1h	RO
06 :04	MME	**Multiple Message Enable:** System software writes to this field to indicate the number of allocated messages (less than or equal to the number of requested messages in MMC). A value of 0 corresponds to one message.		000h	RW
03 :01	MMC	**Multiple Message Capable:** System software reads this field to determine the number of requested messages. Hardwired to 0 to request one message.		000h	RO
00	MSIE	**MSI Enable:** System software sets this bit to enable MSI signaling. A device driver is prohibited from writing this bit to mask a device's service request. If 1, the device can use an MSI to request service. If 0, the device cannot use an MSI to request service.		0h	RW

20.3.3.4 VMSILADDR[0:15]—Message Signalled Interrupt Lower Address Register

Table 20-110.VMSILADDR[0:15]—Message Signalled Interrupt Lower Address Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	**Message Address:** Written by the system to indicate the lower 30-bits of the address to use for the MSI memory write transaction.		0h	RW
01 :00	Reserved	**Reserved**		00b	RV

20.3.3.5 VMSIUADDR[0:15]—Message Signalled Interrupt Upper Address Register

Table 20-111.VMSIUADDR[0:15]—Message Signalled Interrupt Upper Address Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Message Address:** Written by the system to indicate the lower 32-bits of the address to use for the MSI memory write transaction. The lower two bits will always be written as 0.		0h	RW

20.3.3.6 VMSIDATA[0:15]—Message Signalled Interrupt Data Register

Table 20-112.VMSIDATA[0:15]—Message Signalled Interrupt Data Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 9Ch **Offset End:** 9Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	**Message Data:** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWORD transaction. The upper 16 bits of the transaction are written as 0.		0h	RW

20.3.3.7 VMSIMSK—VF Message Signalled Interrupt Mask Register

Table 20-113.VMSIMSK—VF Message Signalled Interrupt Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** M:8 + Index1	**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RV
00 :00	MASK0	**Mask Bits:** Only one bit defined. See the interrupt section.		0b	RW

20.3.3.8 VMSIPND—VF Message Signalled Interrupt Pending Register

Table 20-114.VMSIPND—VF Message Signalled Interrupt Pending Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** M:8 + Index1	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RV
00 :00	Reserved	**Pending Bits:** Only one bit defined. See the interrupt section.		0b	RO

20.3.4 VF Advanced Error Reporting Capability Structure

20.3.4.1 VPCIEAERCAPID[0:15]—VF PCI Express AER Capability ID Register

The PCI Express Capability List register enumerates the PCI Express AER Capability structure in the PCI 3.0 configuration space capability list.

Table 20-115.VPCIEAERCAPID[0:15]—VF PCI Express AER Capability ID Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 100h **Offset End:** 103h	
Size: 32 bit	**Default:** 13810001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 20	PCIEAERNCP	Next PCI Express Extended Capability Pointer: This is hardwired to 138H to point to the Alternative Routing ID extended capability.		138h	RO
19 :16	PCIEAERCVN	Advanced Error Capability Version Number: PCI Express Advanced Error Reporting Extended Capability Version Number.		1h	RO
15 :00	PCIEAERCID	Advanced Error Capability ID: PCI Express Extended Capability ID indicating Advanced Error Reporting Capability.		1h	RO

20.3.4.2 VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register

Table 20-116.VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		00b	RV
20	UR	**Unsupported Request Error Status:** As a receiver, Set whenever an unsupported request is detected. The Header is logged.	Y	0b	RW1CS
19	ECRCC	**ECRC Check:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
18	MTLP	**Malformed TLP:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
17	RO	**Receiver Overflow:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
16	EC	**Unexpected Completion**: As a receiver, set whenever a completion is received that does not match the EP requestor ID or outstanding Tag. The Header is logged.	Y	0b	RW1CS
15	CA	**Completer Abort**: As a completer, set whenever an internal agent signals a data abort. The header is logged.	Y	0b	RW1CS
14	CT	**Completion Timeout:** As a requester, set whenever an outbound Non Posted Request does not receive a completion within 16-32ms.	Y	0b	RW1CS
13	FCPES	**Flow Control Protocol Error Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0b	RV
12	PTLPR	**Poisoned TLP Received:** As a receiver, set whenever a poisoned TLP is received from PCI Express. The header is logged. Internal queue errors are not covered by this bit, they are logged by the Configuration target of the transaction.	Y	0h	RW1CS
11 :6	Reserved	**Reserved.**		0000b	RV

Table 20-116.VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
05	SDES	**Surprise Down Error:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0b	RV
04	DLPE	**Data Link Protocol Error**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0000b	RV
03 00	Reserved	**Reserved**.		0b	RV

20.3.4.3 VPAERUCM[0:15]—VF PCI Express AER Uncorrectable Error Mask Register

Table 20-117.VPAERUCM[0:15]—VF PCI Express AER Uncorrectable Error Mask Register (Sheet 1 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :22	Reserved	Reserved		00b	RV
21	Reserved	**ACS Violation Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		00b	RV
20	UR	**Unsupported Request Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RV
19	ECRCC	**ECRC Check Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
18	MTLP	**Malformed TLP Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV

Table 20-117.VPAERUCM[0:15]—VF PCI Express AER Uncorrectable Error Mask Register (Sheet 2 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17	RO	**Receiver Overflow Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
16	EC	**Unexpected Completion Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RV
15	CA	**Completer Abort Error Mask**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RV
14	CT	**Completion Time Out Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RV
13	FCPES	**Flow Control Protocol Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0b	RV
12	PTLPR	**Poisoned TLP Received Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 20.2.7.3, "PPAERUCM—PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0h	RV
11 :6	Reserved	Reserved.		0000b	RV
05	SDES	**Surprise Down Error Mask**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0b	RV
04	DLPE	**Data Link Protocol Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
03 :01	Reserved	Reserved		0000b	RV
00	Reserved	Reserved		0b	RV

20.3.4.4 VPAERUCSEV[0:15]—VF PCI Express AER Uncorrectable Error Severity Register

Table 20-118.VPAERUCSEV[0:15]—VF PCI Express AER Uncorrectable Error Severity Register (Sheet 1 of 2)

Description:						
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1		**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00h				**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :22	Reserved	Preserved	Preserved		00b	RV
21	AVES	**ACS Violation Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register"applies to all of the VFs.			0b	RV
20	URES	**Unsupported Request Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.			0b	RV
19	ECRCC	**ECRC Check Error Severity**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.			0b	RV
18	MTLP	**Malformed TLP Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.			0b	RV
17	RO	**Receiver Overflow Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.			0b	RV
16	EC	**Unexpected Completion Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.			0b	RV
15	CA	**Completer Abort Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.			0b	RV
14	CT	**Completion Time Out Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.			0b	RV

Table 20-118.VPAERUCSEV[0:15]—VF PCI Express AER Uncorrectable Error Severity Register (Sheet 2 of 2)

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
13	FCPES	**Flow Control Protocol Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0b	RV
12	PTLPR	**Poisoned TLP Received Error Field:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.4, "PPAERUCSEV—PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0h	RV
11 :6	Reserved	Preserved		0000b	RV
5	SDES	**Surprise Down Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros. Not supported.		0000b	RV
04	DLPE	**Data Link Protocol Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
03 :01	Reserved	Preserved		0000b	RV
00	Reserved	**Undefined:** Preserved		0b	RV

20.3.4.5 VPAERCS[0:15]—VF PCI Express AER Correctable Error Status Register

Table 20-119.VPAERCS[0:15]—VF PCI Express AER Correctable Error Status Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	**Reserved**.		00b	RV
13	ANFES	**Advisory Non-Fatal Error Status:**		0b	RV
12	RTTS	**Replay Timer Timeout Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
11 :09	Reserved	**Reserved**.		0b	RV

Table 20-119.VPAERCS[0:15]—VF PCI Express AER Correctable Error Status Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
08	RNRS	**REPLAY NUM Rollover Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0h	RV
07	BDLLPS	**Bad DLLP Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0000b	RV
06	DLPE	**Bad TLP Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV
05 :01	Reserved	**Reserved**.		00000b	RV
00	RES	**Receiver Error Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros.		0b	RV

20.3.4.6 VPAERCM[0:15]—VF PCI Express AER Correctable Error Mask Register

Table 20-120.VPAERCM[0:15]—VF PCI Express AER Correctable Error Mask Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		00b	RV
13	ANFES	**Advisory Non-Fatal Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV
12	RTTS	**Replay Timer Timeout Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV
11 :09	Reserved	Reserved.		000b	RV

Table 20-120.VPAERCM[0:15]—VF PCI Express AER Correctable Error Mask Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
08	RNRS	**REPLAY NUM Rollover Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV
07	BDLLPS	**Bad DLLP Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV
06	DLPE	**Bad TLP Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV
05 :01	Reserved	Reserved.		00h	RV
00	RES	**Receiver Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be treated as Preserved and that the setting of the corresponding bit in the Section 20.2.7.6, "PPAERCM—PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RV

20.3.4.7 VPAERCTLCAP[0:15]—VF PCI Express AER Control and Capability Register

Table 20-121.VPAERCTLCAP[0:15]—VF PCI Express AER Control and Capability Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved		0	RV
08	ECRCCE	**ECRC Check Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.1, "PPCIEAERCAPID—PF PCI Express AER Capability ID Register" applies to all of the VFs. ECRC is not supported		0b	RV
07	ECRCCC	**ECRC Check Capable:** Indicates the EP is **not** capable of checking ECRC.		0b	RO

Table 20-121.VPAERCTLCAP[0:15]—VF PCI Express AER Control and Capability Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	ECRCGE	**ECRC Generation Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 requires that this field be hardwired to all zeros and that the setting of this field in the Section 20.2.7.1, "PPCIEAERCAPID—PF PCI Express AER Capability ID Register" applies to all of the VFs. ECRC is not supported		0b	RV
05	ECRCGC	**ECRC Generation Capable:** Indicates the EP is **not** capable of generating ECRC.		00000b	RO
04 :00	TFEP	**The First Error Pointer:** Identifies the bit position of the first error reported in the Section 20-116, "VPAERUCS[0:15]—VF PCI Express AER Uncorrectable Error Status Register" register. ***Note:*** This register will not update until all bits in the ERRUNC STS register are cleared.	Y	0	ROS-V

20.3.4.8 VPAERHDRLOG0[0:15]—VF PCI Express AER Header Log 0 Register

Table 20-122.VPAERHDRLOG0[0:15]—VF PCI Express AER Header Log 0 Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 11Ch **Offset End:** 11Fh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW0	First DWord of the Header for the PCI Express packet in error (HDRLOGDW0): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.3.4.9 VPAERHDRLOG1[0:15]—VF PCI Express AER Header Log 1 Register

Table 20-123.VPAERHDRLOG1[0:15]—VF PCI Express AER Header Log 1 Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 120h **Offset End:** 123h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW1	2nd DWord of the Header for the PCI Express packet in error (HDRLOGDW1): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.3.4.10 VPAERHDRLOG2[0:15]—VF PCI Express AER Header Log 2 Register

Table 20-124.VPAERHDRLOG2[0:15]—VF PCI Express AER Header Log 2 Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 124h **Offset End:** 127h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW2	3rd DWord of the Header for the PCI Express packet in error (HDRLOGDW2): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.3.4.11 VPAERHDRLOG3[0:15]—VF PCI Express AER Header Log 3 Register

Table 20-125.VPAERHDRLOG3[0:15]—VF PCI Express AER Header Log 3 Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 128h **Offset End:** 12Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRDWLOG3	4th DWord of the Header for the PCI Express packet in error (HDRDWLOG3): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e. the error pointer is rearmed to log again.		0h	RO

20.3.5 VF Alternative Routing ID Extended Capability Structure

This section describes the PCI Express Extended Configuration Space registers that make up the Alternative Routing ID Extended Capability Structure.

Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. See the PCI Express* Base Specification 2.0 for the full register descriptions and additional information regarding their operation.

20.3.5.1 VARIDHDR[0:15]—VF Alternative Routing ID Capability Header

This register contains information associated with the Alternative Routing ID capability. This is compliant with the *PCI-SIG ECN: Alternatifve Routing-ID Interpretation (ARI)*, Updated June 4, 2007.

Table 20-126.VARIDHDR[0:15]—VF Alternative Routing ID Capability Header

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 138h **Offset End:** 13Bh	
Size: 32 bit	**Default:** 1000Eh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ARINCO	**Next Capability Offset**: This field contains 000h indicating the end of the VF Extended capability list.		00h	RO
19 :16	ARICV	**Capability Version**: This is set to 1h for the most current version of the specification.		1h	RO
15 :0	ARICV	**PCI Express Extended Capability ID**: The PCI SIG has assigned 000Eh to the ARI extended capability.		000EH	RO

20.3.5.2 VFARICAP[0:15]—VF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability.

Table 20-127.VFARICAP[0:15]—VF ARI Capabilities Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 0.9 states that this field is undefined for VFs.		0h	RV
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: not supported.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: not supported.		0b	RO

20.3.5.3 VARIDCTL[0:15]—VF Alternative Routing ID Control Register

This register contains information associated with the Alternative Routing ID capability.

Table 20-128.VARIDCTL[0:15]— VF Alternative Routing ID Control Register

Description:					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 13Eh **Offset End:** 13Fh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :7	Reserved	Reserved		0h	RV
6 :4	FG	**Function Group**: Hardwired to Zero.		0b	RO
3 :2	Reserved	Reserved		0b	RV
1	ACS	**ACS Functional Groups Enable**: Hardwired to Zero.		0b	RO
0	MFVC	**MFVC Functional Groups Enable**: Hardwired to Zero.		0b	RO

20.4 EP Memory Mapped Registers

This section describes the MMIO registers that are located in the EP. These registers are exposed from PCI.

20.4.1 Detailed Register Summary

20.4.2 CSRs

This section describes the CSRs that are mapped in PMISCBAR MMIO space.

20.4.2.1 ERRSOU2—Error Source Register 2

This register captures the error sources that are generated by the Ring Controller.

Table 20-129.ERRSOU2—Error Source Register 2

Description: EP Error Sources					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 008h **Offset End:** 00Bh	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :0	BUN150	Bit[n] maps to Interrupt from Bundle[n], where n varies from 0 to 15.	N	0b	RO

20.4.2.2 ERRMSK2—Error Source Mask Register 2

ERRMSK2 provides mask bits for ERRSOU2 interrupts. This register can be used to enable/disable interrupts pending in ERRSOU2. Software can poll ERRSOU2 even if the interrupts are masked.

Table 20-130.ERRMSK2—Error Source Mask Register 2

Description: Error Source Mask 2					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 018h **Offset End:** 01Bh	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :0	BUN150	Mask bits: Bit[n] maps to Interrupt from Bundle[n], where n varies from 0 to 15.	N	0b	RW

20.4.2.3 SINTPF—Signal Raw PF Interrupt Register

This register reflects the EP interrupts for the PF. These interrupts can either be routed to the IA based on values programmed in SMIAPF register. See Section 20.4.2.4, "SMIAPF—Signal IA PF Interrupt Mask Register". The interrupt is sourced from the ERRSOU[2] register.

Table 20-131.SINTPF—Signal Raw PF Interrupt Register

Description: SDATA					
View: PCI	**BAR:** PMISCBAR+1A000h	**Bus:Device:Function:** B:D:0		**Offset Start:** 024h **Offset End:** 027h	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31: 16	Reserved	Reserved		0h	RV
15 :00	BUN0 15	Bundle15 0 Interrupts. See Table 20-129, "ERRSOU2—Error Source Register 2" for interrupts.		0000h	RO

20.4.2.4 SMIAPF—Signal IA PF Interrupt Mask Register

This register controls the ability for the PF to generate an IA interrupt, based on interrupt reflected in the SINTPF register. Each mask bit corresponds to a bit in the SINTPF register. See Table 20-131, "SINTPF—Signal Raw PF Interrupt Register". This register is used to route interrupts reflected in SINTPF to the IA. Software should program SMIAPF register once during system initialization and should not re-program them during normal operation as this register is not meant to be used as interrupt mask bits. In addition, software should program this register appropriately so interrupts are not routed to the IA.

Table 20-132.SMIAPF—Signal IA PF Interrupt Mask Register

Description: SDATA					
View: PCI	**BAR:** PMISCBAR+1A000h	**Bus:Device:Function:** B:D:0		**Offset Start:** 028h **Offset End:** 02Bh	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved		0h	RW
15 :00	BUN0 15	Bundle15 0 Interrupts Mask bit: If set to 1h, an interrupt is sent to the IA as a either an INTx, MSI, or MSI-X. See Table 20-131, "SINTPF—Signal Raw PF Interrupt Register".		0000h	RW

20.4.2.5 GBECFGMMIOV—GBE Configuration and MMIO Valid Register

Since the Configuration and MMIO CSRs are loaded from EEPROM. This register contains four read-only signals, one per GbE port, that indicate when the GbE Configuration and MMIO CSRs are valid. The EP will use these signals and respond with a Configuration Request Retry Status if the Configuration spaces are not valid when the Host tries to access the GbE configuration spaces.

Table 20-133.GBECFGMMIOV - GBE Configuration and MMIO Valid Register

Description: GbE CFG and MMIO Valid.					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 03Ch **Offset End:** 03Fh	
View: PCI PF	**BAR:** PMISCBAR+1A000h		**Bus:Function:** M:0	**Offset Start:** 03Ch **Offset End:** 03Fh	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
:03	GBE3	GbE3:GbE CSRs are loaded from EEPROM this signal indicates when the GbE config and MMIO CSRs are valid. 0 - Valid 1 - Not valid		0b	RO
:02	GBE2	GbE2:GbE CSRs are loaded from EEPROM this signal indicates when the GbE config and MMIO CSRs are valid. 0 - Valid 1 - Not valid		0b	RO
:01	GBE1	GbE1:GbE CSRs are loaded from EEPROM this signal indicates when the GbE config and MMIO CSRs are valid. 0 - Valid 1 - Not valid		0b	RO
:00	GBE0	GbE0:GbE CSRs are loaded from EEPROM this signal indicates when the GbE config and MMIO CSRs are valid. 0 - Valid 1 - Not valid		0b	RO

20.4.2.6 SINTGBE[0:3]—Signal Raw PF Interrupt GBE Register

This register reflects the GbE interrupts. These interrupts can be routed to the IA based on values programmed in SMMEGBE and SMIAGBE registers. See Section 20.4.2.7, "SMIAGBE[0:3]—Signal IA Interrupt Mask GBE Register". These interrupts are sourced from the GbEs.

Table 20-134.SINTGBE[0:3]—Signal Raw PF Interrupt for GBE Register

Description: SDATA					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 040h / at 14h **Offset End:** 043h / at 14h	
Size: 32 bits	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31: 10	Reserved	Reserved		0h	RV
:09	MISCINT	Other causes — Summarizes legacy interrupts into one extended cause.		0h	RO
:08	TCPTIMER	TCP Timer		0h	RO
07 :00	RXTX07	8 Rx/Tx Queue Interrupts		00h	RO

20.4.2.7 SMIAGBE[0:3]—Signal IA Interrupt Mask GBE Register

This register controls the ability for the GBE to generate an IA interrupt, based on pending interrupt the SINTGBE register. Each mask bit corresponds to a bit in the SINTGBE register. See Table 20-134, "SINTGBE[0:3]—Signal Raw PF Interrupt for GBE Register". This register is used to route interrupts reflected in SINTGBE to either IA. Software should program SMIAGBE register once during system initialization and should not re-program this register during normal operation as thesethis register is not meant to be used as interrupt mask bits. In addition, software should program this register appropriately so interrupts are not routed to the IA.

Table 20-135.SMIAGBE[0:3]—Signal IA PF Interrupt Mask GBE Register

Description: SDATA					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 044h / at 14h **Offset End:** 047h / at 14h	
Size: 32 bits	**Default:** 000003FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved		0h	RW

Table 20-135.SMIAGBE[0:3]—Signal IA PF Interrupt Mask GBE Register

Description: SDATA					
View: PCI	**BAR:** PMISCBAR+1A000h		**Bus:Device:Function:** B:D:0	**Offset Start:** 044h / at 14h **Offset End:** 047h / at 14h	
Size: 32 bits	**Default:** 000003FFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:09	MISCINT	MISCINT from SINTGBE. When this bit is set, an interrupt is sent to the IA as either an INTx, MSI, or MSI-X. Default interrupt steered to IA.		1b	RW
:08	TCPTIMER	TCP Timer from SINTGBE. When this bit is set, an interrupt is sent to the IA as either an INTx, MSI, or MSI-X. Default interrupt steered to IA.		1b	RW
07 :00	RXTX07	8 Rx/Tx Queue Interrupts. 1-1 mapping to SINTGBE bits. When a bit is set, an interrupt is sent to the IA as either an INTx, MSI, or MSI-X. Default interrupts steered to IA.		FFh	RW

Each VINTSOUx is an interrupt pending register which corresponds to a given VFx. The register can be used by software to identify the source of the VFx interrupt. A VINTSOUx register is mapped in both PF and its corresponding VFx address space. Access to the PF is provided only for testing or debugging purposes. Interrupts in the VINTSOUx can be masked using the its corresponding VINTMSKx register.

Each VINTMSKx corresponds to a VINTSOUx register. Each VINTMSKx corresponds a given VFx. Software can use the VINTMSKx to enable and disable interrupts pending in VINSOUx.

§ §

21.0 GbE Controller Overview

The PCH Gigabit Ethernet (GbE) Controller provides four fully-integrated Gigabit Ethernet Media Access Control (MAC) controllers with four SGMII/SerDes ports that can be connected to an external PHY. The GbE Controller can support up to four SerDes ports LAN on Motherboard (LOM) design or mezzanine card. The GbE Controller can also be used in embedded applications such as add-on card switches and network appliances.

Note: The GbE Controller is SKU dependant.

- The PCH GbE Ethernet Interface and the integrated four GbE MAC Controllers, are collectively referred to as the "GbE Controller" or the "Controller" throughout this document.

21.1 Feature Summary

External interfaces provided:

- Serializer-Deserializer (SerDes) to support 1000Base-SX/LX (optical fiber).
- Serializer-Deserializer (SerDes) to support 1000BASE-KX and 1000BASE-BX for Gigabit backplane applications.
- SGMII interface for SFP/external PHY[1] connections
- SMBus for Manageability connection to BMC

Intel® I/O Acceleration Technology v3.0 is supported

- Stateless offloads (Header split, RSS)
- Direct Cache Access

Power saving features:

- Advanced Configuration and Power Interface (ACPI) power management states and wake-up capability
- Advanced Power Management (APM) wake-up functionality

Additional product details:

- Support for PCI 3.0 vital product data
- Eight TX and eight RX queues
- IPMI BMC pass-thru
- DMTF Management Component Transport Protocol (MCTP) over SMBus/ I^2C and MCTP over PCIe* Vendor Defined Messaging
- IEEE 1588 Precision Time Protocol support
- Per-Packet timestamp

1. See the "*Supported Ethernet PHY Devices for the Intel® Communications Chipset 89xx Series*" Application Note.

21.2 Scope

This document provides the external architecture (including device operation, pin descriptions, register definitions, etc.) for the GbE interface in.

This document is a reference for software device driver developers, board designers, test engineers, and others who may need specific technical or programming information.

21.3 Terminology and Acronyms

Table 21-1. Glossary

Term	Definition
1000BASE-BX	1000BASE-BX is the PICMG 3.1 electrical specification for transmission of 1 Gb/s Ethernet or 1 Gb/s fibre channel encoded data over the backplane.
1000BASE-KX	1000BASE-KX is the IEEE802.3ap electrical specification for transmission of 1 Gb/s Ethernet over the backplane.
1000BASE-CX	1000BASE-X over specialty shielded 150 Ω balanced copper jumper cable assemblies as specified in IEEE 802.3 Clause 39.
1000BASE-T	1000BASE-T is the specification for 1 Gb/s Ethernet over category 5e twisted pair cables as defined in IEEE 802.3 clause 40.
b/w	Bandwidth.
BIOS	Basic Input/Output System.
BMC	Baseboard Management Controller.
BT	Bit Time.
CGB	Control Interface for the GbE Controller on the EP Bus
DCA	Direct Cache Access.
DFT	Design for Testability.
DQ	Descriptor Queue.
DW	Double word (4 bytes).
EEPROM	Electrically Erasable Programmable Read Only Memory. A non-volatile memory located on the LAN controller that is directly accessible from the host.
EOP	End of Packet.
FC	Flow Control.
Firmware (FW)	Embedded code on the LAN controller that is responsible for the implementation of the pass through functionality.
Host Interface	RAM on the LAN controller that is shared between the firmware and the host. RAM is used to pass commands from the host to firmware and responses from the firmware to the host.
HPC	High Performance Computing.
IPC	Inter Processor Communication.
IPG	Inter Packet Gap.
LAN (auxiliary Power-Up)	The event of connecting the LAN controller to a power source (occurs even before system power-up).
LOM	LAN on Motherboard.
LTR	Latency Tolerance Reporting (PCIe* protocol)
LSO	Large Send Offload.
MAC	Media Access Control.

Table 21-1. Glossary (Continued)

Term	Definition
MDIO	Management Data Input/Output Interface over MDC/MDIO lines.
MIFS/MIPG	Minimum Inter Frame Spacing/Minimum Inter Packet Gap.
MMW	Maximum Memory Window.
MSS	Maximum Segment Size. Largest amount of data, in a packet (without headers) that can be transmitted. Specified in Bytes.
MPS	Maximum Payload Size in PCIe* specification.
MTU	Maximum Transmit Unit. Largest packet size (headers and data) that can be transmitted. Specified in bytes.
NIC	Network Interface Controller.
OBFF	Opportunistic Buffer Flush/Fill (PCIe* protocol).
PBA	Printed Board Assembly
PCS	Physical Coding Sub layer.
PHY	Physical Layer Device.
PMA	Physical Medium Attachment.
PMD	Physical Medium Dependent.
RMII	Reduced Media Independent Interface (Reduced MII).
SA (In MAC context)	Source Address.
SDP	Software Defined Pins.
SerDes	Serializer and De-Serializer Circuit.
SFD	Start Frame Delimiter.
SGMII	Serialized Gigabit Media Independent Interface.
SMBus	System Management Bus. A bus that carries various manageability components, including the LAN controller, BIOS, sensors and remote-control devices.
TCO	Total Cost of Ownership (TCO) System Management.
TLP	Transaction Layer Packet in the PCI Express specification.
TPH	TLP Process Hints (PCIe* protocol).
TSO	Transmit Segmentation offload - A mode in which a large TCP/UDP I/O is handled to the device and the device segments it to L2 packets according to the requested MSS.
VPD	Vital Product Data (PCI protocol).

21.3.1 External Specification and Documents

The GbE Controller implements features from the following specifications.

21.3.1.1 Network Interface Documents

- IEEE standard 802.3, 2006 Edition (Ethernet). Incorporates various IEEE Standards previously published separately. Institute of Electrical and Electronic Engineers (IEEE).
- IEEE standard 1149.1, 2001 Edition (JTAG). Institute of Electrical and Electronics Engineers (IEEE)
- IEEE Std 1149.6-2003, IEEE Standard for Boundary-Scan Testing of Advanced Digital Networks, IEEE, 2003.
- IEEE standard 802.1Q for VLAN
- PICMG3.1 Ethernet/Fibre Channel Over PICMG 3.0 Draft Specification, January 14, 2003, Version D1.0
- Serial-GMII Specification, Cisco Systems document ENG-46158, Revision 1.7
- INF-8074i Specification for SFP (Small Form factor Pluggable) Transceiver (ftp://ftp.seagate.com/sff)
- IEEE Std 802.3ap-2007
- IEEE 1588* Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, November 8 2002
- IEEE 802.1AS Timing and Synchronization for Time- Sensitive Applications in Bridged Local Area Networks Draft 2.0, February 22, 2008

21.3.1.2 Host Interface Documents

- PCI-Express 2.0 Base specification, Revision 1.0
- PCI Specification, version 3.0
- PCI Bus Power Management Interface Specification, Rev. 1.2, March 2004
- Advanced Configuration and Power Interface Specification, Rev 2.0b, October 2002

21.3.1.3 Networking Protocol Documents

- IPv4 specification (RFC 791)
- IPv6 specification (RFC 2460)
- TCP/UDP specification (RFC 793/768)
- SCTP specification (RFC 2960)
- ARP specification (RFC 826)
- EUI-64 specification, http://standards.ieee.org/regauth/oui/tutorials/EUI64.html.

21.3.1.4 Manageability Document

System Management Bus (SMBus) Specification, SBS Implementers Forum, Ver. 2.0, August 2000

21.4 Product Overview

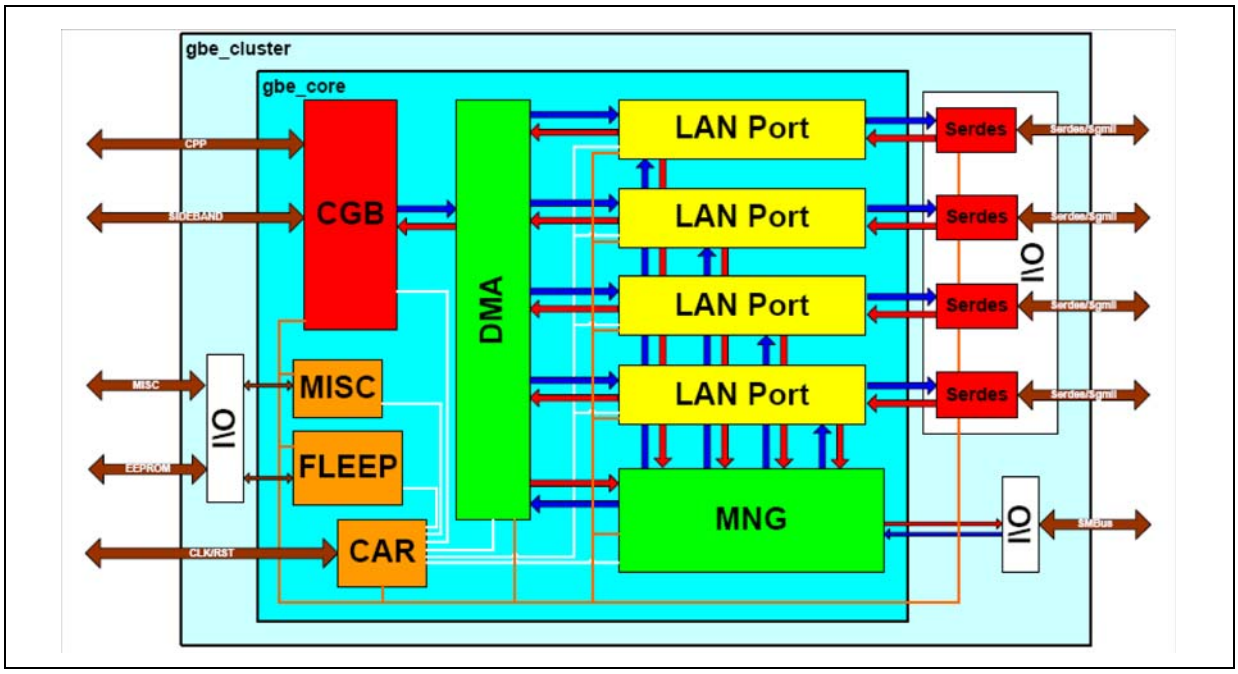
The Controller supports four SerDes or SGMII ports for MAC to external PHY connections.

21.5 External Interface

Figure 21-1 shows the high-level connectivity of the GbE interface in a PCH implementation.

Note: FLEEP means Flash/EEPROM module.

Figure 21-1. High-Level View of GbE Interface Connectivity in PCH Implementation



21.5.1 PCIe* Interface (Connected to PCI Express Through PCH PCIe* End Point)

The PCIe* Gen2 Interface, provided by PCH PCIe* Endpoint implementation, is used by PCH as a host interface. The four GbE ports of the Controller behave like four separate PCI functions in End Point Configuration.

21.5.2 Network Interfaces

Four independent interfaces are used to connect the four Controller ports to external devices. The following protocols are supported:

- SerDes interface to connect over a backplane to another SerDes compliant device or to an Optical module. The Controller supports both 1000BASE-BX and 1000BASE-KX (Without IEEE802.3ap Backplane Auto-Negotiation).
- SGMII interface to attach to an external PHY, either on board or via an SFP module. The SGMII interface shares the same pins as the SerDes.

21.5.3 EEPROM Interface

The EEPROM stores information needed for the GbE MACs and other PCH functionality. This other functionality mandates an EEPROM necessary for PCH even if the GbE MACs are not being used. However, if all GbE MACs are disabled via the EEPROM, then the IA core will not be able to access the EEPROM.

The Controller uses an EEPROM device for storing product configuration information. Several words of the EEPROM are accessed automatically by the Controller after reset in order to provide pre-boot configuration data that must be available to PCH before it is accessed by host software. The remainder of the stored information is accessed by various software modules used to report product configuration, serial number, etc.

The Controller is intended for use with a SPI (4-wire) serial EEPROM device such as an AT25040AN or compatible EEPROM device.

21.5.4 SMBus Interface

SMBus is an optional interface for pass-through and/or configuration traffic between a BMC and the Controller.

The Controller's SMBus interface can be configured to support both slow and fast timing modes.

21.5.5 MDIO/I^2C Two-Wire Interfaces

The Controller implements four management Interfaces for control of an external PHY. Each interface can be either a 2-wire Standard-mode I^2C interface used to control an SFP module or an MII Management Interface (also known as the Management Data Input/Output or MDIO Interface) for control plane connection between the MAC and PHY devices (master side). This interface provides the MAC and software with the ability to monitor and control the state of the external PHY. The Controller supports the data formats defined in IEEE 802.3 clause 22.

The Controller supports shared MDIO operation for all ports or separate MDIO connection per port. When configured via the *MDICNFG* register to separate MDIO operation each MDIO interface should be connected to the relevant PHY. When configured via the *MDICNFG* register to shared MDIO operation the MDC/MDIO interface of LAN port 0 can be shared by all ports to support connection to a multi-port PHY with a single MDC/MDIO interface.

21.5.6 Software-Definable Pins (SDP) Interface (General-Purpose I/O)

The Controller has two software-defined pins (SDP pins) per port that can be used for miscellaneous hardware or software-control purposes. These pins can be individually configurable to act as either input or output pins. The default direction of each pin is configurable via the EEPROM (see Section 24.3.13 and Section 28.1.4.2), as well as the default value of all pins configured as outputs. To avoid signal contention, all pins are set as input pins until the EEPROM configuration is loaded. All SDP pins can be configured for use as general-purpose interrupt (GPI) inputs. To act as GPI pins, the desired pins must be configured as inputs. A corresponding GPI interrupt-detection enable bit is then used to enable rising-edge detection of the input pin (rising-edge detection occurs by comparing values sampled at the internal clock rate, as opposed to an edge-detection circuit). When detected, a corresponding GPI interrupt is indicated in the Interrupt Cause register.

The use, direction, and values of SDP pins are controlled and accessed using fields in the Device Control (CTRL) register and Extended Device Control (CTRL_EXT) register.

21.5.7 LED Interface

The Controller implements one output driver per port intended for driving external LED circuits. LED outputs can be individually configured to select the particular event, state, or activity, which is indicated on that output. In addition, each LED can be individually configured for output polarity as well as for blinking versus non-blinking (steady-state) indication.

The configuration for LED outputs is specified via the LEDCTL register. Furthermore, the hardware-default configuration for all LED outputs can be specified via EEPROM fields (see Section 24.3.12 and Section 24.3.12), thereby supporting LED displays configurable to a particular OEM preference.

See Section 26.5 for more detailed description of LED behavior.

21.6 GbE Controller Features and Capabilities

Table 21-2 through Table 21-7 list the Controller's features and capabilities.

Table 21-2. GbE Controller Features

Feature	GbE Controller
Serial FLASH interface	N
4-wire SPI EEPROM interface	Y
Configurable LED operation for software or OEM custom-tailoring of LED displays	Y
Protected EEPROM space for private configuration	N
Watchdog timer	Y

Table 21-3. GbE Controller Network Features

Feature	GbE Controller
Half duplex at 10/100 Mb/s operation and full duplex operation at all supported speeds	Y
Jumbo frames supported	Y
Size of jumbo frames supported	9.5 KB
Flow control support: send/receive PAUSE frames and receive FIFO thresholds	Y
Statistics for management and RMON	Y
802.1q VLAN support	Y
SerDes interface for external PHY connection or system interconnect	4 ports
1000BASE-KX interface for Blade Server Backplane connections	Y
802.3ap Backplane Auto-negotiation	N
SGMII interface for external 1000BASE-T PHY connection	4 ports
SerDes support of non-Auto-Negotiation partner	Y
SerDes signal detect	Y
External PHY control I/F MDC/MDIO 2 wire I/F	Shared or per function Per function
I^2C clock stretching	Y

Table 21-4. GbE Controller Host Interface Features

Feature	GbE Controller
64-bit address support for systems using more than 4 GB of physical memory	Y
Outstanding requests for Tx buffers per port	24 Per port and for all ports
Outstanding requests for Tx descriptors per port	4 Per port and for all ports
Outstanding requests for Rx descriptors per port	4 Per port and for all ports
Max payload size supported	256 Bytes
Max request size supported	1 KB
Vital Product Data (VPD)	Y
64-bit CSR access using 64-bit operation[1]	N
CSR access via Configuration space	Y

1. 64-bit CSRs should be access using 2 seperate 32-bit read/write operations.

Table 21-5. GbE Controller LAN Functions Features

Feature	GbE Controller
Programmable host memory receive buffers	Y
Descriptor ring management hardware for transmit and receive	Y
ACPI register set and power down functionality supporting D0 & D3 states	Y
IEEE802.3az (Energy Efficient Ethernet)	N
Software controlled global reset bit (resets everything except the configuration registers)	Y
Software Definable Pins (SDP) - per port	2
SDP pins can be configured as general purpose interrupts	Y
Wake up	Y
Flexible wake-up filters	8
Flexible filters for queue assignment in normal operation	8
IPv6 wake-up filters	Y
Default configuration by the EEPROM for all LEDs for pre-driver functionality	1 LED
LAN function disable capability	Y
Programmable memory transmit buffers	Y
Double VLAN	Y
IEEE 1588	Y
Per-Packet Timestamp	Y

Table 21-6. GbE Controller LAN Performance Features

Feature	GbE Controller
TCP segmentation offload Up to 256 KB	Y
IPv6 support for IP/TCP and IP/UDP receive checksum offload	Y
Fragmented UDP checksum offload for packet reassembly	Y
Message Signaled Interrupts (MSI)	Y
Message Signaled Interrupts (MSI-X) number of vectors per port	10
Packet interrupt coalescing timers (packet timers) and absolute-delay interrupt timers for both transmit and receive operation	Y
Interrupt throttling control to limit maximum interrupt rate and improve CPU utilization	Y
Rx packet split header	Y
Receive Side Scaling (RSS) number of queues per port	Up to 8
Total number of Rx queues per port	8
Total number of TX queues per port	8
RX header replication Low latency interrupt DCA support (Legacy DCA support only) TCP timer interrupts No snoop Relax ordering	Yes to all
TSO interleaving for reduced latency	Y
Receive side coalescing	N
SCTP receive and transmit checksum offload	Y
UDP TSO	Y
IPSec offload	N

Table 21-7. GbE Controller Virtualization Features

Feature	GbE Controller
Support for Virtual Machines Device queues (VMDq) per port	8 pools (single queue)
L2 MAC address filters (unicast and multicast)	24
L2 VLAN filters	Per pool
PCI-SIG SR-IOV	N
Multicast/Broadcast Packet replication	Y on Receive
VM to VM Packet forwarding (Packet Loopback)	N
RSS replication	N
Traffic shaping	N
MAC and VLAN anti-spoofing	N
Malicious driver detection	Y
Per-pool statistics	Y

Table 21-7. GbE Controller Virtualization Features (Continued)

Feature	GbE Controller
Per-pool off loads	Y
Per-pool jumbo support	Y
Mirroring rules	4

21.6.1 Network Interface

21.6.1.1 Quad Port

The Controller supports four LAN ports (10/100/1000BASE-T on SerDes) and associated MAC and DMA queues enabling reduction of BOM cost and board space.

21.6.1.2 Shared MDIO Support

The Controller enables sharing of the MDIO interface on LAN port 0 by all ports connected to an external 1000BASE-T PHY to support connection to an external multi-port PHY device.

To abstract actual MDIO interface configuration from Software driver, MDIO setup (PHY Address and Shared MDIO) is loaded into the MDICNFG register from the EEPROM following reset. Further information can be found in Section 22.5.2.2 and Section 28.1.4.6.

21.6.1.3 I^2C Clock Stretching

There are situations where an I^2C slave can not support the clock speed or needs to delay an access initiated by the master. Delaying the access is done by a mechanism referred to as clock stretching. An I^2C slave is allowed to hold down the clock if it needs to delay an access. The master is required to read back the clock signal after releasing it to a high-z state and wait until the line has actually gone high as a result of an external Pull-up resistor.

When configured to operate in I^2C mode, the Controller supports clock stretching on the I2C_CLK pin. When accessing PHY registers via the I^2C interface the PHY can delay the access or reduce clock speed if required.

21.6.1.4 1000BASE-KX Backplane Ethernet Interface

The Controller can interface up to four 1Gbps 1000BASE-KX lanes for Blade Server backplane interconnect without need for additional glue logic. The Controller supports only parallel detection of 1000BASE-KX signaling and does not support the full auto-negotiation for Backplane Ethernet protocol.

21.6.2 HOST Interface

21.6.2.1 PCIe* Connectivity Through EndPoint Interface

The Controller connects to PCIe* through a resident End Point as four functions of that EndPoint.

21.6.2.2 64-bit BAR Support

The Controller supports 64-bit BAR.

21.6.3 Performance Features

21.6.4 Receive and Transmit Queues

The number of Receive and Transmit queues supported by the Controller for Receive Side Scaling (RSS) filtering and Virtualization is 8 per port.

21.6.4.1 Unused Receive and Transmit Ports Buffer Sharing

The Controller supports up to four Gigabit Ethernet ports. When device is configured to work only as a single port or dual port device, available internal receive and transmit buffer memory for the remaining active ports, can be increased by four for the single port case and by two for the dual port case. See Section 26.1.3.2 and Section 26.2.1.2 for more information.

21.6.5 Virtualization

The Controller does not support the following virtualization features:

- PCIe* SR-IOV
- VM to VM packet forwarding
- Anti-Spoof protection

The Controller supports 8 queues per port that can be shared between eight virtual machines (VMDq2). By associating receive and transmit packets to separate queues dedicated to a VM, performance of the VMM (Virtual Machine Monitor) is improved. The Controller also supports the following hardware Virtualization features:

- Per VM interrupt assignment
- Enabling read of statistic counters by VMs without initiating a clear by read
- Copy of received multicast and broadcast packets to multiple queues
- Per VM Offload
- Per Pool statistics
- Mirroring
- Storm Control

21.6.5.1 Malicious Driver Detection

Malicious behavior exhibited by the driver can be a result of incorrect activation of the network controller or a virus on a certain VM. The Controller contains internal circuitry to protect from an attack on one virtual machine disrupting operation of other virtual machines. When malicious driver behavior is detected on a certain queue, PCH disables activity of the queue and sends notification to the VMM. See Section 26.8.3.7.2 for details.

21.6.5.2 2-tuple Filtering

The Controller supports only 2-tuple filtering compared to previous Intel products that supported 5-tuple filtering. The Controller enables queuing, generating immediate interrupts and time stamping according to TCP/UDP port destination address and L4 protocol fields. See Chapter 26.0, "GbE Inline Functions" for more information.

21.6.6 Security Offload

The Controller does not support the IPSec offload features.

21.6.6.1 Transmit Queue Prioritization

The Controller supports strict Transmit queue prioritization to enable transmission of high priority real-time packets ahead of low priority packets. A Transmit queue is either a high priority queue or low priority queue, with the high priority queues always served ahead of the low priority queues. See Section 26.2.2.5 for information.

21.6.7 Manageability

21.6.7.1 SMBus Interface to External BMC

The Controller supports four managed ports on a single SMBus interface to external BMC.

21.6.7.1.1 Teaming and Fail-over Support by BMC

The LAN ports act independently of each other and each port has a separate SMBus address. As a result, fail-over where manageability traffic is routed to an active port if any of the ports fail, is not supported internally. The BMC is responsible for teaming and fail-over.

21.6.7.1.2 Auto-ARP Reply on SMBus

The Controller cannot be programmed for auto-ARP reply on reception of ARP request packets and does not support sending of gratuitous ARP packets to reduce the traffic over the BMC interconnect.

21.6.7.2 Exclusive Management Filtering

In the Controller, decisions regarding forwarding of packets to the host and to the BMC are separate and are configured through two sets of registers. However, the BMC may define some types of traffic as exclusive. This traffic will be forwarded only to the BMC, even if it passes the filtering process of the host. These types of traffic are defined using the MNGONLY register. Further information can be found in Section 26.1.2.3 and Section 27.4.1.

21.6.8 Time SYNC (IEEE1588 and IEEE 802.1AS)

21.6.8.1 Per Packet Timestamp

The Controller supports adding an optional tailored header before the MAC header of the packet in the receive buffer. The 128 bit tailored header contains a 64 bit timestamp (MSB bits) and 64 reserved bits (LSB bits).

The timestamp is sampled on the MAC/PHY interface, on detection of a received packet that meets certain criteria. Time stamping the received packet on the PHY/MAC interface avoids incurring additional delays due to PCIe* latencies. The timestamp is placed in the receive buffer ahead of the actual received packet. See "Inline Functions/Receive Packet Timestamp in Buffer" section for more details.

21.6.8.2 Improved System Time Accuracy

System time accuracy has been increased in the Controller. System time corrections with a resolution of 2^{-32} ns can be entered using the *TIMINCA.Incvalue* field.

21.6.8.3 SYSTIM Synchronized Pulse Generation on SDP Pins

The Controller has the capability to generate a pulse synchronized to system time.

21.6.8.4 Time SYNC Interrupts

The Controller has the additional capability to generate interrupts on occurrence of Time Sync events.

21.7 Device Data Flows

21.7.1 Transmit Data Flow

Tx data flow provides a high level description of all data/control transformation steps needed for sending Ethernet packets to the line.

Table 21-8. Transmit Data Flow

Step	Description
1	The host creates a descriptor ring and configures one of PCH's transmit queues with the address location, length, head and tail pointers of the ring (one of 16 available Tx queues).
2	The host is requested by the TCP/IP stack to transmit a packet, it gets the packet data within one or more data buffers.
3	The host initializes descriptor(s) that point to the data buffer(s) and have additional control parameters that describe the needed hardware functionality. The host places that descriptor in the correct location at the appropriate Tx ring.
4	The host updates the appropriate queue tail pointer (TDT)
5	PCH's DMA senses a change of a specific TDT and as a result sends a request to fetch the descriptor(s) from host memory.
6	The descriptor(s) content is received in a PCIe* read completion and is written to the appropriate location in the descriptor queue internal cache.
7	The DMA fetches the next descriptor from the internal cache and processes its content. As a result, the DMA sends requests to fetch the packet data from system memory.
8	The packet data is received and passes through the transmit DMA that performs all programmed data manipulations (various CPU off loading tasks as checksum off load, TSO off load, etc.) on the packet data on the fly.
9	While the packet is passing through the DMA, it is stored into the transmit FIFO. After the entire packet is stored in the transmit FIFO, it is forwarded to the transmit switch module.
10	The transmit switch arbitrates between host and management packets and eventually forwards the packet to the MAC.
11	The MAC appends the L2 CRC to the packet and sends the packet to the line using a pre-configured interface.
12	When all the completions for a given packet are done, the DMA updates the appropriate descriptor(s).
13	After enough descriptors are gathered for write back or the interrupt moderation timer expires, the descriptors are written back to host memory using writes transactions (through EP interface). Alternatively, the head pointer can only be written back.
14	After the interrupt moderation timer expires, an interrupt is generated to notify the host device driver that the specific packet has been read to PCH and the driver can release the buffers.

21.7.2 Rx Data Flow

Rx data flow provides a high level description of all data/control transformation steps needed for receiving Ethernet packets.

Table 21-9. Receive Data Flow

Step	Description
1	The host creates a descriptor ring and configures one of PCH's receive queues with the address location, length, head, and tail pointers of the ring (one of 16 available Rx queues).
2	The host initializes descriptors that point to empty data buffers. The host places these descriptors in the correct location at the appropriate Rx ring.
3	The host updates the appropriate queue tail pointer (RDT).
4	PCH's DMA senses a change of a specific RDT and as a result sends a request to fetch the descriptors from host memory.
5	The descriptors content is received in a read completion (using EP interface) and is written to the appropriate location in the descriptor queue internal cache.
6	As packet enters the Rx MAC. The RX MAC checks the CRC of the packet.
7	The MAC forwards the packet to an Rx filter
8	If the packet matches the pre-programmed criteria of the Rx filtering, it is forwarded to the Rx FIFO. VLAN and CRC are optionally stripped from the packet and L3/L4 checksum are checked and the destination queue is fixed.
9	The receive DMA fetches the next descriptor from the internal cache of the appropriate queue to be used for the next received packet.
10	After the entire packet is placed into the Rx FIFO, the receive DMA posts the packet data to the location indicated by the descriptor through the EP interface. If the packet size is greater than the buffer size, more descriptors are fetched and their buffers are used for the received packet.
11	When the packet is placed into host memory, the receive DMA updates all the descriptor(s) that were used by packet data.
12	After enough descriptors are gathered for write back or the interrupt moderation timer expires or the packet requires immediate forwarding, the receive DMA writes back the descriptor content along with status bits that indicate the packet information including what off loads were done on that packet.
13	After the interrupt moderation timer completes or an immediate packet is received, PCH initiates an interrupt to the host to indicate that a new received packet is already in host memory.
14	Host reads the packet data and sends it to the TCP/IP stack for further processing. The host releases the associated buffers and descriptors once they are no longer in use.

§ §

22.0 GbE Interconnects

22.1 PCIe* Interface

The GbE interface connects to PCIe* through the EndPoint, therefore all PCIe* related transaction details as well as error handling are explained in the PCIe* EndPoint section of this document.

22.1.1 Special GbE Controller Features

- Packet sizes/formats:
  - Maximum upstream (write) PCIe* payload size of 256 bytes
  - Maximum downstream (read) PCIe* payload size of 1K bytes
- Baseline messaging:
  - In-band messaging of formerly side-band legacy signals (such as interrupts, etc.)
  - System-level power management supported via messages
- Power management:
  - Full support for PCI-PM
  - Wake capability from D3cold state
  - Compliant with ACPI, PCI-PM software model
  - Active state power management

22.1.2 Relaxed Ordering

The controller takes advantage of the relaxed ordering rules in PCIe*. By setting the relaxed ordering bit in the packet header, the controller enables the system to optimize performance in the following cases:

- Relaxed ordering for descriptor and data reads: When the controller emits a read transaction, its split completion has no ordering relationship with the writes from the CPUs (same direction). It should be allowed to bypass the writes from the CPUs.
- Relaxed ordering for receiving data writes: When the controller issues receive DMA data writes, it also enables them to bypass each other in the path to system memory because software does not process this data until their associated descriptor writes complete.
- The controller cannot relax ordering for descriptor writes, MSI/MSI-X writes or PCIe* messages.

Relaxed ordering can be used in conjunction with the no-snoop attribute to enable the memory controller to advance non-snoop writes ahead of earlier snooped writes.

Relaxed ordering is enabled in the controller by clearing the *RO_DIS* bit in the CTRL_EXT register. Actual setting of relaxed ordering is done for LAN traffic by the host through the DCA registers.

22.1.3 Snoop Not Required

The controller sets the *Snoop Not Required* attribute bit for master data writes. System logic might provide a separate path into system memory for non-coherent traffic. The non-coherent path to system memory provides higher, more uniform, bandwidth for write requests.

Note: The *Snoop Not Required* attribute does not alter transaction ordering. Therefore, to achieve maximum benefit from *Snoop Not Required* transactions, it is advisable to set the relaxed ordering attribute as well (assuming that system logic supports both attributes). In fact, some chipsets require that relaxed ordering is set for no-snoop to take effect.

Global no-snoop support is enabled in the controller by clearing the *NS_DIS* bit in the *CTRL_EXT* register. Actual setting of no snoop is done for LAN traffic by the host through the DCA registers.

22.1.4 No Snoop and Relaxed Ordering for LAN Traffic

Software might configure non-snoop and relax order attributes for each queue and each type of transaction by setting the respective bits in the RXCTRL and TXCTRL registers.

Table 22-1 lists the default behavior for the *No-Snoop* and *Relaxed Ordering* bits for LAN traffic when I/OAT 2 is enabled.

Table 22-1. LAN Traffic Attributes

Transaction	No-Snoop Default	Relaxed Ordering Default	Comments
Rx Descriptor Read	N	Y	
Rx Descriptor Write-Back	N	N	Relaxed ordering must never be used for this traffic.
Rx Data Write	Y	Y	See the note below.
Rx Replicated Header	N	Y	
Tx Descriptor Read	N	Y	
Tx Descriptor Write-Back	N	Y	
Tx TSO Header Read	N	Y	
Tx Data Read	N	Y	

Note: Rx payload no-snoop is also conditioned by the *NSE* bit in the receive descriptor. See Section 22.1.4.1.

22.1.4.1 No-Snoop Option for Payload

Under certain conditions, which occur when I/OAT is enabled, software knows that it is safe to transfer (DMA) a new packet into a certain buffer without snooping. This scenario typically occurs when software is posting a receive buffer to hardware that the CPU has not accessed since the last time it was owned by hardware. This might happen if the data was transferred to an application buffer by the I/OAT DMA engine.

In this case, software should be able to set a bit in the receive descriptor indicating that the controller should perform a no-snoop DMA transfer when it eventually writes a packet to this buffer.

When a non-snoop transaction is activated, the TLP header has a non-snoop attribute in the *Transaction Descriptor* field.

This is triggered by the *NSE* bit in the receive descriptor.

22.1.5 PCIe* Power Management

Described in Chapter 25.0, "GbE Power Management".

22.2 Management Interfaces

The controller contains one SMBus interface to an external BMC for manageability.

- SMBus

Since the manageability sideband throughput is lower than the network link throughput, the controller allocates a 2 KB internal buffer for incoming network packets prior to being sent over the sideband interface. The controller also allocates a 2 KB internal buffer for outgoing network packets prior to being sent over the Ethernet link.

Allocated buffer size is a function of number of active LAN ports. If only 2 LAN ports are enabled, internal Buffer size is 4 KB. If only a single LAN port is enabled, internal buffer size is 8 KB. Enabled LAN ports are defined in the *Manageability Capability / Manageability Enable* EEPROM word (Word 0x54).

22.2.1 SMBus

SMBus is an optional interface for pass-through and/or configuration traffic between an external BMC and the controller. The SMBus commands used to configure or read status from the controller are described in Chapter 27.0, "GbE Platform Manageability".

22.2.1.1 Channel Behavior

22.2.1.1.1 SMBus Addressing

The SMBus is presented as four SMBus devices on the SMBus were each device has a different SMBus addresses. All pass-through functionality is duplicated per SMBus address, where each SMBus address is connected to a different LAN port.

Note: DO NOT configure ports to the same address. When a LAN function is disabled, the corresponding SMBus address is not presented to the external BMC.

The SMBus addresses are set by *SMBus Address 0*, *SMBus Address 1, SMBus Address 2* and *SMBus Address 3* words in the EEPROM.

The SMBus addresses (those that are enabled from the EEPROM) can be re-assigned using the SMBus ARP protocol.

Besides the SMBus address values, all the previously stated parameters of the SMBus (SMBus channel selection, address mode, address enable) can be set only through EEPROM configuration. The EEPROM is read by the controller at power-up, resets, and other cases described in Section 23.2, "Reset Operation".

All SMBus addresses should be in Network Byte Order (NBO); most significant byte first.

22.2.1.1.2 SMBus Notification Methods

The controller supports three methods of informing the external BMC that it has information that is needed to be read by an external BMC:

- SMBus alert.
- Asynchronous notify.
- Direct receive.

The notification method that is used by the controller can be configured from the SMBus using the Receive Enable command. The default method is set from the EEPROM in the *Notification Method* field.

The following events cause the controller to send a notification event to the external BMC:

- Receiving a LAN packet that was designated to the BMC.
- Receiving a request status command from the BMC initiates a status response (see Chapter 27.0, "GbE Platform Manageability").
- Status change has occurred and the controller is configured to notify the external BMC upon one of the status changes. The following event triggers a notification to the BMC: A change in any of the *Status Data 1* bits of the Read Status command (see Chapter 27.0, "GbE Platform Manageability" for description of this command).

There might be cases where the external BMC is hung and is unable to respond to the SMBus notification. The controller has a time-out value defined in the EEPROM (see Section 24.0, "Non-Volatile Memory Map - EEPROM") to avoid hanging while waiting for the notification response. If the BMC does not respond before the timeout expires, the notification is de-asserted.

22.2.1.1.2.1 SMBus Alert and Alert Response Method

The SMBus Alert# signal acts as an asynchronous interrupt signal to an external SMBus master. The controller asserts this signal each time it has a message that it needs the external BMC to read and if the chosen notification method is the SMBus-alert method. The SMBus alert is an open-drain signal, which means that other devices besides the controller can be connected on the same alert pin and the external BMC needs a mechanism to distinguish between the alert sources as described:

The external BMC can respond to the alert by issuing an ARA (Alert Response Address) cycle (see Figure 22-2) to detect the alert source device. The controller responds to the ARA cycle (if it was the SMBus alert source) and de-asserts the alert when the ARA cycle completes. Following the ARA cycle, the external BMC issues a Read command to retrieve the controller message.

Some BMCs do not implement ARA cycle transactions. These BMCs respond to an alert by issuing a Read command to all active controller ports (0xC0/0xD0 or 0xDE). The controller always responds to a Read command, even if it is not the source of the notification. The default response is a status transaction. If the controller is the source of the SMBus alert, it replies to the read transaction and de-asserts the alert after the command byte of the read transaction.

The ARA cycle is an SMBus receive byte transaction to SMBus Address 0001-100b. The ARA transaction does not support PEC. The ARA transaction format is as follows:

Table 22-2. SMBus ARA Cycle Format

1	7	1	1	7	1	1	1
S	Alert Response Address	Rd	A	Slave Device Address		A	P
	0001 100	1	0	Manageability Slave SMBus Address	0	1	

Note: Since the master-receiver (BMC receiver) is involved in the transaction it must signal the end of data to the PCH by generating a NACK (a '1' in the ACK bit position) on the Slave Device Address byte that was clocked out by the PCH. The PCH releases the data line to allow the master to generate a STOP condition.

22.2.1.1.2.2 Asynchronous Notify Method

When configured to asynchronous notify method, the controller acts as SMBus master and notifies the external BMC by issuing a modified form of the write word transaction. The asynchronous notify transaction SMBus address and data payload is configured using the Receive Enable command or using the EEPROM defaults. The asynchronous notify method is not protected by a PEC byte.

Table 22-3. Asynchronous Notify Command Format

1	7	1	1	7	1	1	
S	Target Address	Wr	A	Sending Device Address		A	•••
	BMC Slave Address	0	0	Manageability Slave SMBus Address	0	0	

8	1	8	1	1
Data Byte Low	A	Data Byte High	A	P
Interface	0	Alert Value	0	

The target address and data byte low/high is taken from the Receive Enable command (see Chapter 27.0, "GbE Platform Manageability") or EEPROM configuration (See Section 24.0, "Non-Volatile Memory Map - EEPROM").

22.2.1.1.2.3 Direct Receive Method

If configured, the controller has the capability to send the message it needs to transfer to the external BMC as a master over the SMBus, instead of alerting the BMC, and waiting for it to read the message.

Table 22-4 shows the message format when receiving a LAN packet that was designated to the BMC. The "F", "L" and command fields in the message are the same as the op-code returned by the PCH in response to a BMC *Receive TCO Packet* Block read command (See Chapter 27.0, "GbE Platform Manageability"). The rules for the "F" and "L" flags are also the same as used in the *Receive TCO Packet* Block Read command.

Table 22-4. Direct Receive - LAN Packet Receive Transaction Format

1	7	1	1	1	1	6	1	
S	Target Address	Wr	A	F	L	Command	A	•••
	BMC Slave Address	0	0	First Flag	Last Flag	Receive TCO Command 01 0000b	0	

8	1	8	1		1	8	1	1
Byte Count	A	Data Byte 1	A	•••	A	Data Byte N	A	P
N	0		0		0		0	

The following table shows the message format when a status change has occurred and the controller is configured to notify the external BMC upon a status change. The op-code and Status data fields returned by the controller are the same as in response to a BMC *Read Status* command (see Chapter 27.0, "GbE Platform Manageability").

Table 22-5. Direct Receive - Status Change Transaction Format

1	7	1	1	8	1	
S	Target Address	Wr	A	Command	A	•••
	BMC Slave Address	0	0	Read status op-code 0xDD	0	

8	1	8	1	8	1	1
Byte Count	A	Data 2 (Status Data 1)	A	Data 3 (Status Data 2)	A	P
0x02	0		0		0	

22.2.1.1.3 Receive TCO Flow

The controller is used as a channel for receiving packets from the network link and passing them to the external BMC. The BMC can configure the controller to pass specific packets to the BMC. Once a full packet is received from the link and identified as a manageability packet that should be transferred to the BMC, the controller starts the receive TCO transaction flow to the BMC.

The maximum SMBus fragment length is defined in the EERPOM. The controller uses the SMBus notification method to notify the BMC that it has data to deliver. The packet is divided into fragments, where the controller uses the maximum fragment size allowed in each fragment. The last fragment of the packet transfer is always the status of the packet. As a result, the packet is transferred in at least two fragments. The data of the packet is transferred in the Receive TCO LAN packet transaction.

When SMBus alert is selected as the BMC notification method, the controller notifies the BMC on each fragment of a multi-fragment packet. When asynchronous notify is selected as the BMC notification method, the controller notifies the BMC only on the first fragment of a received packet. It is BMC's responsibility to read the full packet including all the fragments.

Any timeout on SMBus notification results in discarding the entire packet. Any NACK by the BMC on one of the controller receive bytes also causes the packet to be silently discarded.

If a SMBus time-out occurs during reception of a packet from the network to the BMC, the controller silently discards the packet.

The maximum size of the received packet is limited by the controller hardware to 1536 bytes. Packets larger then 1536 bytes are silently discarded. Any packet smaller than 1536 bytes is processed by the controller.

Note: When the *RCV_EN* bit is cleared, all receive TCO functionality is disabled, not just the packets that are directed to the BMC.

22.2.1.1.4 Transmit TCO Flow

The controller is used as a channel for transmitting packets from the external BMC to the network link. The network packet is transferred from the external BMC over the SMBus, and then, when fully received by the controller, is transmitted over the network link.

Each SMBus address is connected to a different LAN port. When a packet is received in SMBus transactions using *SMBus Address 0*, *SMBus Address 1*, *SMBus Address 2* or *SMBus Address 3* it is transmitted to the network using LAN port 0, LAN port 1, LAN port 2 or LAN port 3 respectively.

The controller supports packets up to the Ethernet packet length (1536 bytes). SMBus transactions can be up to 240 bytes in length, which means that packets can be transferred over the SMBus in more than one fragment. In each command byte there are the *F* and *L* bits. When the *F* bit is set, it means that this is the first fragment of the packet; *L* means that it is the last fragment of the packet.

Note: When both flags are set, the entire packet is in one fragment.

The packet is sent over the network link, only after all its fragments are received correctly over the SMBus.

The controller calculates the L2 CRC on the transmitted packet and adds its four bytes at the end of the packet. Any other packet field (such as XSUM) must be calculated and inserted by the external BMC (The controller does not change any field in the transmitted packet, besides adding padding and CRC bytes).

Note: If the packet sent by the BMC is larger than 1536 bytes, then the packet is discarded by the controller and Abort is asserted.

The minimum packet length defined by the 802.3 specification is 64 bytes. The controller pads packets that are less than 64 bytes to meet the specification requirements. There is one exception, when the packet sent over the SMBus is less than 32 bytes, the external BMC must pad it for at least 32 bytes. The passing bytes value should be zero.

Note: Packets that are smaller than 32 bytes (including padding), are discarded by the controller and Abort is asserted.

If the network link goes down at anytime while the controller is receiving a packet from the BMC for transmission on the network, it silently discards the packet. Any link down event during the transfer of a packet to the BMC over the SMBus (after it is fully received from the network), does not stop the operation.

Note: If a SMBus time-out occurs during transmission of a packet from the BMC to the network the controller silently discards the packet.

The transmit SMBus transactions are described in Chapter 27.0, "GbE Platform Manageability".

22.2.1.1.4.4 Transmit Errors in Sequence Handling

Once a packet is transferred over the SMBus from the BMC to the controller, the *F* and *L* flags should follow specific rules. The *F* flag defines that this is the first fragment of the packet; The *L* flag defines that the transaction contains the last fragment of the packet.

The following table lists the options regarding the flags in transmit packet transactions:

Table 22-6. Flags in Transmit Packet Transactions

Previous	Current	Action/Notes
Last	First	Accepts both.
Last	Not First	Error for current transaction. Current transaction is discarded and an abort status is asserted.
Not Last	First	Error for previous transaction. Previous transaction (until previous first) is discarded. Current packet is processed. No abort status is asserted.
Not Last	Not First	Processes the current transaction.

Since every other Block Write command in the TCO protocol has both *F* and *L* flags off, they cause flushing any pending transmit fragments that were previously received. In other words, when running the TCO transmit flow, no other block write transactions are allowed in between the fragments.

22.2.1.1.5 TCO Command Aborted Flow

Bit 6 in first byte of the status returned from the controller to the external BMC indicates that there was a problem with previous SMBus transactions or with the completion of the operation requested in previous transaction.

An abort can be asserted for any of the following reasons:

- Any error in the SMBus protocol (NACK, SMBus timeout).
- If the BMC does not respond until the notification timeout (programmed in the EEPROM) expires.
- Any error in compatibility between required protocols to specific functionality (Receive Enable command with byte count not 1/14 as defined in the command specification).
- If the controller does not have space to store the transmit packet from the BMC (in its internal buffer before sending it to the link). In this case, the entire transaction completes, but the packet is discarded and the BMC is notified about it through the *Abort* bit.
- Error in the *F*/*L* bit sequence during multi-fragment transactions.
- If the packet sent by the BMC is larger than 1536 bytes.
- If the packet sent by the BMC is smaller than 32 bytes (including padding).
- An internal reset to the controller manageability unit occurred.
- Following an unsuccessful internal register access when the *MCSR_TO_RETRY* EEPROM bit is cleared.

Note: An abort in the status does not always imply that the last transaction of the sequence was incorrect. There is a gap between the time the status is read from the controller and the time the transaction occurred.

22.2.1.1.6 Concurrent SMBus Transactions

Concurrent SMBus transactions (receive, transmit and configuration read/write) are allowed between the four addresses supported by the controller. Transmit fragments can be sent between receive fragments and configuration Read/Write commands can also be issued between receive and transmit fragments.

22.2.1.1.7 SMBus ARP Functionality

The controller supports SMBus ARP protocol as defined in the SMBus 2.0 specification. The controller is a persistent slave address device meaning that its SMBus address is valid after power-up and loaded from the EEPROM. The controller supports all SMBus ARP commands defined in the SMBus specification, both general and directed.

Note: SMBus ARP can be disabled through EEPROM configuration (See Chapter 27.0, "GbE Platform Manageability").

SMBus-ARP transactions are described in Chapter 27.0, "GbE Platform Manageability".

22.2.1.1.7.5 SMBus ARP Response Behavior

The controller responds as four SMBus devices, meaning that it has four sets of *AR*/*AV* flags (one for each port). The controller responds four times to the SMBus-ARP master, one time for each port. All SMBus addresses are taken from the SMBus ARP addresses word of the EEPROM. The UDID is different between the four ports in the Vendor Specific ID field, which represent the MAC address, which is different between the four ports. The controller answers first as port 0, and only when the address is assigned, starts answering as port 1, 2, and 3 to the Get UDID command.

22.2.1.1.7.6 SMBus ARP Flow

SMBus-ARP flow is based on the status of two flags:

- *AV* - Address Valid - This flag is set when the controller has a valid SMBus address.
- *AR* - Address Resolved - This flag is set when the controller's SMBus address is resolved (SMBus address was assigned by the SMBus-ARP process).

Note: These flags are internal the controller flags and not shown to external SMBus devices.

Since the controller is a Persistent SMBus Address (PSA) device, the *AV* flag is always set, while the *AR* flag is cleared after power-up until the SMBus-ARP process completes. Since the *AV* flag is always set, the PCH always has a valid SMBus address.

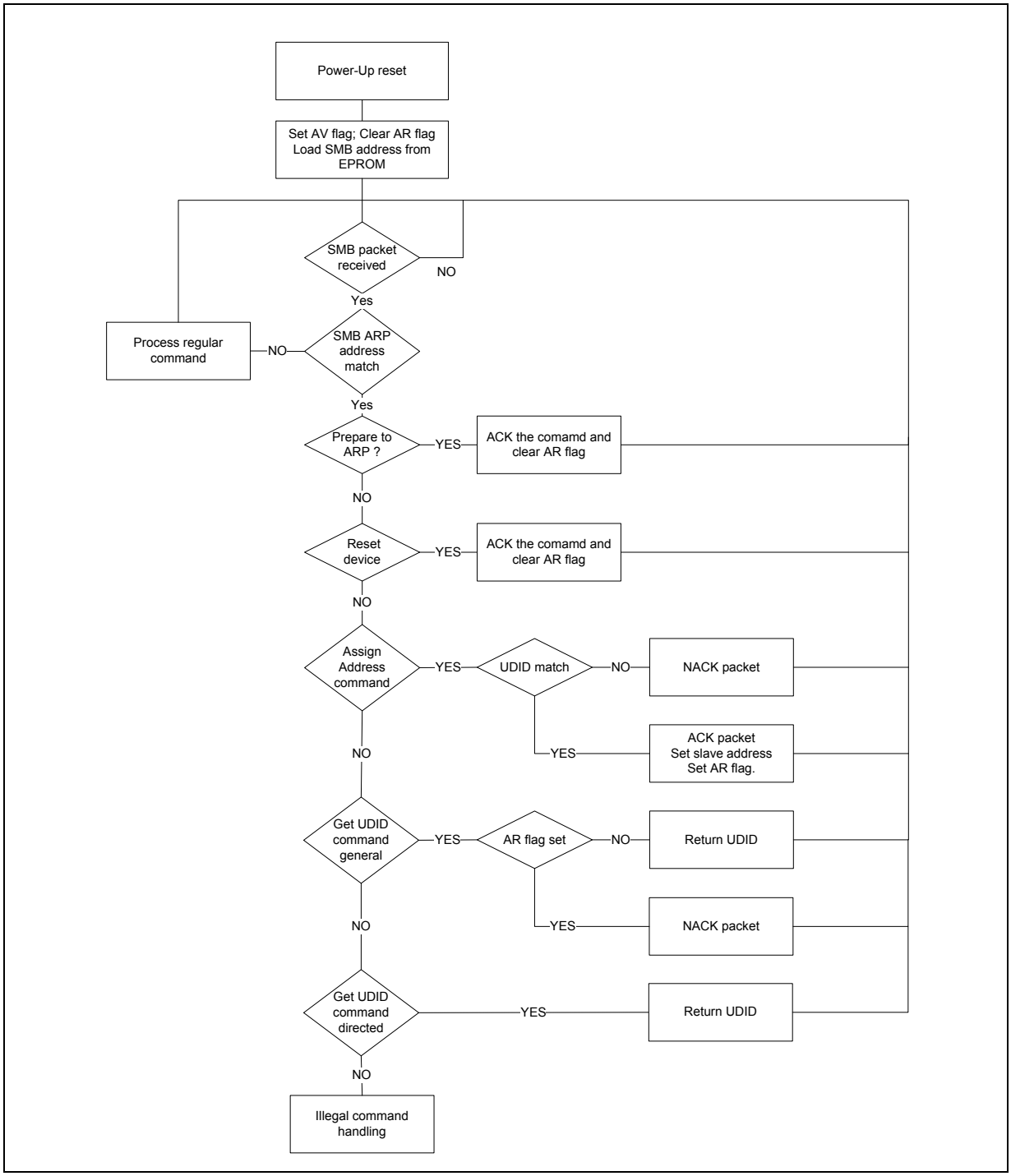
When the SMBus master needs to start an SMBus-ARP process, it resets (In terms of ARP functionality) all the devices on the SMBus by issuing either Prepare to ARP or Reset Device commands. When the controller accepts one of these commands, it clears its *AR* flag (if set from previous SMBus-ARP process), but not its *AV* flag (the current SMBus address remains valid until the end of the SMBus ARP process).

The meaning of an *AR* flag cleared is that the controller answers the following SMBus ARP transactions that are issued by the master. The SMBus master then issues a Get UDID command (general or directed), to identify the devices on the SMBus. The controller responds to the directed command all the time and to the general command only if its *AR* flag is not set. After the Get UDID command, the master assigns the controller's SMBus address by issuing an Assign Address command. The controller

checks whether the UDID matches its own UDID, and if they match, it switches its SMBus address to the address assigned by the command (byte 17). After accepting the Assign Address command, the *AR* flag is set and from this point on (as long as the *AR* flag is set), the controller does not respond to the Get UDID general command, while all other commands should be processed even if the *AR* flag is set. The controller stores the SMBus address that was assigned in the SMBus-ARP process in its EEPROM, so after the next power-up, it returns to its assigned SMBus address.

Figure 22-7 shows the SMBus-ARP behavior of the controller.

Figure 22-7. SMBus ARP Flow



22.2.1.1.7.8 SMBus ARP UDID Content

The Unique Device Identifier (UDID) provides a mechanism to isolate each device for the purpose of address assignment. Each device has a unique identifier. The 128-bit number is comprised of the following fields:

Table 22-7. Unique Device Identifier (UDID)

1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes	2 Bytes	2 Bytes	4 Bytes
Device Capabilities	Version / Revision	Vendor ID	Device ID	Interface	Sub-system Vendor ID	Sub- system Device ID	Vendor Specific ID
See below	See below	0x8086	PCIe Dev ID	0x0004	0x0000	0x0000	See below
MSB							LSB

Where:

- Vendor ID - The device manufacturer's ID as assigned by the SBS Implementers' Forum or the PCI SIG - Constant value: 0x8086.
- Device ID - The device ID as assigned by the device manufacturer (identified by the *Vendor ID* field) - Constant value
- Interface - Identifies the protocol layer interfaces supported over the SMBus connection by the device - In this case, SMBus Version 2.0 - Constant value: 0x0004.
- Sub-system Fields - These fields are not supported and return zeros.

Device Capabilities: Dynamic and Persistent Address, PEC Support bit:

Table 22-8. Dynamic and Persistent Address, PEC Support Bit

7	6	5	4	3	2	1	0
Address Type		Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	PEC Supported
0b	1b	0b	0b	0b	0b	0b	0b
MSB							LSB

Version/Revision: UDID Version 1, Silicon Revision:

Table 22-9. Version/Revision: UDID Version 1, Silicon Revision

7	6	5	4	3	2	1	0
Reserved (0)	Reserved (0)	UDID Version			Silicon Revision ID		
0b	0b	001b			See below		
MSB							LSB

Table 22-10. Silicon Revision ID

Silicon version	Revision ID
A0	000b
B0	001b
C0	010b
C1	011b

Vendor Specific ID - Four LSB bytes of the controller Ethernet MAC address. The controller Ethernet address is taken from offsets 0 to 1 from start of the relevant sections in the EEPROM.

In the PCH there are four MAC addresses (one for each port).

Table 22-11. Vendor Specific ID

1 Byte	1 Byte	1 Byte	1 Byte
MAC Address, byte 3	MAC Address, byte 2	MAC Address, byte 1	MAC Address, byte 0
MSB			LSB

22.3 EEPROM

22.3.1 EEPROM Interface

22.3.1.1 General Overview

The PCH uses an EEPROM device for storing product configuration information. The EEPROM is divided into three general regions:

- Hardware accessed - Loaded by the controller after power-up, PCI reset de-assertion, (D3 ->D0 transition, or a software-commanded EEPROM read (CTRL_EXT.EE_RST).
- Manageability firmware accessed - Loaded by the controller in pass-through mode after power-up or firmware reset.
- Software accessed - Used only by software. The meaning of these registers, as listed here, is a convention for software only and is ignored by the controller.

Table 22-12 lists the structure of the EEPROM image in the PCH

Table 22-12. EEPROM Structure

Address	Content
0x0 – 0x9	LAN 0 MAC address and software area
0xA – 0x2F	LAN 0 and Common hardware area
0x30 – 0x3E	PXE area
0x3F	Software Checksum, for Words 0x00 - 0x3F
0x40 – 0x4F	Software area
0x50 – 0x7F	FW pointers
0x80 -0xBF	LAN 1 hardware area (with SW checksum in 0xBF)
0xC0 - 0xFF	LAN 2 hardware area (with SW checksum in 0xFF)
0x100 - 0x13F	LAN 3 hardware area (with SW checksum in 0x13F)

The EEPROM mapping is described in Section 24.0, “Non-Volatile Memory Map - EEPROM”.

22.3.1.2 EEPROM Device

The EEPROM interface supports an SPI interface and expects the EEPROM to be capable of 2 MHz operation.

The controller is compatible with various sizes of 4-wire serial EEPROM devices. If pass-through mode functionality is desired, up to 256 Kbits serial SPI compatible EEPROM can be used. If no manageability mode is desired, a 8192-bit serial SPI compatible EEPROM can be used. All EEPROM's are accessed in 16-bit words although the EEPROM is designed to also accept 8-bit data accesses.

The controller automatically determines the address size to be used with the SPI EEPROM it is connected to and sets the EEPROM address size field of the *EEPROM Control and Data* register (*EEC.EE_ADDR_SIZE*) field appropriately. Software can use this size to determine how to access the EEPROM. The exact size of the EEPROM is determined within one of the EEPROM words.

Note: The different EEPROM sizes have two differing numbers of address bits (8 bits or 16 bits), and therefore must be accessed with a slightly different serial protocol. Software must be aware of this if it accesses the EEPROM using direct access.

22.3.1.3 HW Initial Load Process

Upon power on reset or PCIe* reset, the controller reads the global device parameters from the EEPROM including all the parameters impacting the content of the PCIe* configuration space. Upon a software reset to one of the ports (*CTRL.RST* set to 1), a partial load is done of the parameters relevant to the port where the software reset occurred. Upon a software reset to all ports (*CTRL.DEV_RST* = 1) a partial load is done of the parameters relevant to all ports. Table 22-13 lists the words read in each EEPROM auto-read sequence. During full load after power-on all hardware related EEPROM words are loaded. Following a software reset only a subset of the hardware related EEPROM words are loaded. For details of the content of each word - see Section 24.0, “Non-Volatile Memory Map - EEPROM”.

LANx_start parameter in Table 22-13 relates to start of LAN related EEPROM section where:

- LAN0_start = 0x0
- LAN1_start = 0x80
- LAN2_start = 0xC0
- LAN3_start = 0x100

Table 22-13. EEPROM Auto-Load Sequence (Sheet 1 of 4)

EEPROM Word	EEPROM Word Address	Full Load (Power-up)	Full Load No MGMT (PCI RST)	SW[1] reset port 0 Load	SW[1] reset port 1 Load	SW[1] reset port 2 Load	SW[1] reset port 3 Load
EEPROM sizing	012	Y	Y	Y	Y	Y	Y
CSR Auto Configuration Power-Up LAN0	027	Y					
CSR Auto Configuration Power-Up LAN1	LAN1_start + 027	Y					
CSR Auto Configuration Power-Up LAN2	LAN2_start + 027	Y					

Table 22-13. EEPROM Auto-Load Sequence (Sheet 2 of 4)

EEPROM Word	EEPROM Word Address	Full Load (Power-up)	Full Load No MGMT (PCI RST)	SW[1] reset port 0 Load	SW[1] reset port 1 Load	SW[1] reset port 2 Load	SW[1] reset port 3 Load
CSR Auto Configuration Power-Up LAN3	LAN3_start + 027	Y					
Init Control 1	00A	Y	Y	Y	Y	Y	Y
Device Rev ID	01E	Y	Y				
Subsystem ID[2]	00B	Y	Y				
Subsystem Vendor ID[2]	00C	Y	Y				
Device ID - LAN 0[3]	00D	Y	Y				
Device ID - LAN 1[3]	LAN1_start + 00D	Y	Y				
Device ID - LAN 2[3]	LAN2_start + 00D	Y	Y				
Device ID - LAN 3[3]	LAN3_start + 00D	Y	Y				
Vendor ID - LAN 0[3]	00E	Y	Y				
Phy_Rst_Control0	016	Y	Y				
Phy_Rst_Control1	LAN1_start + 016	Y	Y				
Phy_Rst_Control2	LAN2_start + 016	Y	Y				
Phy_Rst_Control3	LAN3_start + 016	Y	Y				
LAN power consumption	022	Y	Y				
CSR Auto Configuration Pointer and CSR Auto Configuration structures - LAN0	017	Y	Y	Y			
CSR Auto Configuration Pointer and CSR Auto Configuration structures - LAN1	LAN1_start + 017	Y	Y		Y		
CSR Auto Configuration Pointer and CSR Auto Configuration structures - LAN2	LAN2_start + 017	Y	Y			Y	
CSR Auto Configuration Pointer and CSR Auto Configuration structures - LAN3	LAN3_start + 017	Y	Y				Y
VPD Pointer to table	02F	Y	Y				
VPD table entry ID TAG	ID STRING	Y	Y				
VPD read or write area TAG	VPD TAG 1	Y	Y				
VPD read or write area length	VPD TAG 1 LENGTH	Y	Y				
VPD read or write area TAG	VPD TAG 2	Y	Y				
VPD read or write area length	VPD TAG 2 LENGTH	Y	Y				
VPD end TAG	VPD END	Y	Y				
Init Control 3 LAN 0	024	Y	Y				
Init Control 3 LAN 1	LAN1_start + 024	Y	Y				

Table 22-13. EEPROM Auto-Load Sequence (Sheet 3 of 4)

EEPROM Word	EEPROM Word Address	Full Load (Power-up)	Full Load No MGMT (PCI RST)	SW[1] reset port 0 Load	SW[1] reset port 1 Load	SW[1] reset port 2 Load	SW[1] reset port 3 Load
Init Control 3 LAN 2	LAN2_start + 024	Y	Y				
Init Control 3 LAN 3	LAN3_start + 024	Y	Y				
Init Control 4 LAN 0	013	Y	Y	Y			
Init Control 4 LAN 1	LAN1_start + 013	Y	Y		Y		
Init Control 4 LAN 2	LAN2_start + 013	Y	Y			Y	
Init Control 4 LAN 3	LAN3_start + 013	Y	Y				Y
LEDCTL 0 default LAN 0	01F	Y	Y				
LEDCTL 0 default LAN 1	LAN1_start + 01F	Y	Y				
LEDCTL 0 2 default LAN 2	LAN2_start + 01F	Y	Y				
LEDCTL 0 2 default LAN 3	LAN3_start + 01F	Y	Y				
Init Control 2	00F	Y	Y	Y	Y	Y	Y
Ethernet address byte 2-1 - LAN 0	000	Y	Y	Y			
Ethernet address byte 4-3 - LAN 0	001	Y	Y	Y			
Ethernet address byte 6-5 - LAN 0	002	Y	Y	Y			
Ethernet address byte 2-1 - LAN 1	LAN1_start + 000	Y	Y		Y		
Ethernet address byte 4-3 - LAN 1	LAN1_start + 001	Y	Y		Y		
Ethernet address byte 6-5 - LAN 1	LAN1_start + 002	Y	Y		Y		
Ethernet address byte 2-1 - LAN 2	LAN2_start + 000	Y	Y			Y	
Ethernet address byte 4-3 - LAN 2	LAN2_start + 001	Y	Y			Y	
Ethernet address byte 6-5 - LAN 2	LAN2_start + 002	Y	Y			Y	
Ethernet address byte 2-1 - LAN 3	LAN3_start + 000	Y	Y				Y
Ethernet address byte 4-3 - LAN 3	LAN3_start + 001	Y	Y				Y
Ethernet address byte 6-5 - LAN 3	LAN3_start + 002	Y	Y				Y
Software defined pins control - LAN0	020	Y	Y	Y			
Software defined pins control - LAN1	LAN1_start + 020	Y	Y		Y		
Software defined pins control - LAN2	LAN2_start + 020	Y	Y			Y	
Software defined pins control - LAN3	LAN3_start + 020	Y	Y				Y
Management Section[4]							
Pass Through LAN Configuration Pointer LAN0	011	Y					

Table 22-13. EEPROM Auto-Load Sequence (Sheet 4 of 4)

EEPROM Word	EEPROM Word Address	Full Load (Power-up)	Full Load No MGMT (PCI RST)	SW[1] reset port 0 Load	SW[1] reset port 1 Load	SW[1] reset port 2 Load	SW[1] reset port 3 Load
Pass Through LAN Configuration Pointer LAN1	LAN1_start + 011	Y					
Pass Through LAN Configuration Pointer LAN2	LAN2_start + 011	Y					
Pass Through LAN Configuration Pointer LAN3	LAN3_start + 011	Y					
Management Hardware Config Control	023	Y					
MNG Capabilities	054	Y					
Sideband Configuration Pointer	057	Y					
Firmware patch Pointer	051	Y					
HW Section Continued							
Watchdog configuration	02E[5]	Y	Y	Y	Y	Y	Y

1. Upon assertion of *CTRL.DEV_RST* by software partial load of parameters relevant to all ports is done. Assertion of *CTRL_EXT.EE_RST* causes load of per port parameters similar to *CTRL.RST*.
2. Loaded only if load subsystem ID bit is set
3. Loaded only if load device ID bit is set
4. EEPROM words listed under Management Section are also loaded following Firmware Reset
5. Word also loaded following Firmware reset.

22.3.1.4 Software Accesses

The controller provides two methods for software access to the EEPROM. It can either use the built-in controller to read the EEPROM or access the EEPROM directly using the EEPROM's 4-wire interface.

Software can use the EEPROM Read (*EERD*) register to cause the controller to read a word from the EEPROM that the software can then use. To do this, software writes the address to read to the *Read Address* (*EERD.ADDR*) field and simultaneously writes a 1b to the *Start Read* bit (*EERD.START*). The controller reads the word from the EEPROM, sets the *Read Done* bit (*EERD.DONE*), and places the data in the *Read Data* field (*EERD.DATA*). Software can poll the EEPROM Read register until it sees the *Read Done* bit set and then uses the data from the *Read Data* field. Any words read this way are not written to the PCH's internal registers.

Software can also directly access the EEPROM's 4-wire interface through the EEPROM Control and Data (*EEC*) register. It can use this for reads, writes, or other EEPROM operations.

To directly access the EEPROM, software should follow these steps:

1. Write a 1b to the *EEPROM Reques*t bit (*EEC.EE_REQ*).
2. Read the *EEPROM Grant* bit (*EEC.EE_GNT*) until it becomes 1b. It remains 0b as long as the hardware is accessing the EEPROM.
3. Write or read the EEPROM using the direct access to the 4-wire interface as defined in the EEPROM Control and Data (*EEC*) register. The exact protocol used depends on the EEPROM placed on the board and can be found in the appropriate datasheet.
4. Write a 0b to the *EEPROM Request* bit (*EEC.EE_REQ*) to enable EEPROM access by other drivers.

Finally, software can cause the controller to re-read the per-function hardware accessed fields of the EEPROM (setting the controller's internal registers appropriately similar to software reset) by writing a 1b to the *EEPROM Reset* bit of the Extended Device Control register (*CTRL_EXT.EE_RST*).

Note: If the EEPROM does not contain a valid signature (see Section 22.3.1.5), the controller assumes 16-bit addressing. In order to access an EEPROM that requires 8-bit addressing, software must use the direct access mode.

22.3.1.5 Signature Field

The controller determines if an EEPROM is present by attempting to read it. The controller first reads the *EEPROM Sizing* word at address 0x12. It checks the signature value for bits 15 and 14. If bit 15 is 0b and bit 14 is 1b, it considers the EEPROM to be present and valid and reads additional EEPROM words and then programs its internal registers based on the values read. Otherwise, it ignores the values it reads from that location and does not read any other words as part of the auto-read process. However, the EEPROM is still accessible to software.

In most applications, initial EEPROM programming is done directly on the EEPROM pins. Nevertheless, it is desired to enable existing software utilities (accessing the EEPROM via the host interface) to initially program the entire EEPROM. Following a power-up sequence, the controller reads the hardware initialization words in the EEPROM. If the signature in word 0x12 does not equal 01b, the EEPROM is assumed as non-programmed. There are two effects of a non-valid signature:

- The controller does not read any further EEPROM data.
- The controller enables access to any location in the EEPROM via the EEPROM CSR registers.

22.3.1.6 EEPROM Recovery

The PCH requires EEPROM to boot and therefore must be provided with valid EEPROM before power up.

22.3.2 Shared EEPROM

The controller uses a single EEPROM device to configure hardware default parameters for all LAN devices, including Ethernet Individual Addresses (IA), LED behaviors, receive packet filters for manageability, wake-up capability, etc. Certain EEPROM words are used to specify hardware parameters that are LAN device-independent (such as those that affect circuit behavior). Other EEPROM words are associated with a specific LAN device. All LAN devices access the EEPROM to obtain their respective configuration settings.

22.3.2.1 EEPROM Deadlock Avoidance

The EEPROM is a shared resource between the following clients:

- Hardware auto-read.
- Port 0 LAN driver accesses.
- Port 1 LAN driver accesses.
- Port 2 LAN driver accesses.
- Port 3 LAN driver accesses.
- Firmware accesses.

All clients can access the EEPROM using parallel access, where hardware implements the actual access to the EEPROM. Hardware can schedule these accesses so that all clients get served without starvation.

However, software and hardware clients can access the EEPROM using bit banging. In this case, there is a request/grant mechanism that locks the EEPROM to the exclusive usage of one client. If this client is stuck (without releasing the lock), the other clients are not able to access the EEPROM. In order to avoid this, the controller implements a timeout mechanism, which releases the grant from a client that didn't toggle the EEPROM bit-bang interface for more than two seconds. The EEPROM deadlock avoidance mechanism is enabled when the *Deadlock Timeout Enable* bit in the *Initialization Control Word 1* EEPROM word is set to 1.

Note: If an agent that was granted access to the EEPROM for bit-bang access didn't toggle the bit bang interface for 500 ms, it should check if it still owns the interface before continuing the bit-banging.

22.3.2.2 EEPROM Map Shared Words

The EEPROM map in Chapter 24.0, "Non-Volatile Memory Map - EEPROM" identifies those words configuring either LAN devices or the entire controller Quad Port Interface as "all". Those words configuring a specific LAN device parameter are identified by their LAN number.

The following EEPROM words warrant additional notes specifically related to quad-LAN support:

Table 22-14. Notes on EEPROM Words

Initialization Control 1, Initialization Control 2 (shared between LANs)	These EEPROM words specify hardware-default values for parameters that apply a single value to all LAN devices, such as link configuration parameters required for auto-negotiation, wake-up settings, PCIe* bus advertised capabilities, etc.
Initialization Control 3, Initialization Control 4 (unique to each LAN)	This EEPROM word configures default values associated with each LAN device's hardware connections, including which link mode (SGMII, SerDes, 1000BASE-KX) is used with this LAN device. Because a separate EEPROM word configures the defaults for each LAN, extra care must be taken to ensure that the EEPROM image does not specify a resource conflict.

22.3.3 Vital Product Data (VPD) Support

The EEPROM image might contain an area for VPD. This area is managed by the OEM vendor and doesn't influence the behavior of hardware. Word 0x2F of the EEPROM image contains a pointer to the VPD area in the EEPROM. A value of 0xFFFF means VPD is not supported and the VPD capability doesn't appear in the configuration space.

The VPD area should be aligned to a Dword boundary in the EEPROM.

The maximum area size is 256 bytes but can be smaller. The VPD block is built from a list of resources. A resource can be either large or small. The structure of these resources is listed in the following tables.

Table 22-15. Small Resource Structure

Offset	0	1 - n
Content	Tag = 0xxx, xyyyb (Type = Small(0), Item Name = xxxx, length = yyy bytes)	Data

Table 22-16. Large Resource Structure

Offset	0	1 - 2	3 - n
Content	Tag = 1xxx, xxxxb (Type = Large(1), Item Name = xxxxxxx)	Length	Data

The controller parses the VPD structure during the auto-load process (power up and PCIe* reset or warm reset) in order to detect the read-only and read/write area boundaries. The controller assumes the following VPD structure:

Table 22-17. VPD Structure

Tag	Structure Type	Length (Bytes)	Data	Resource Description
0x82	Large	Length of identifier string	Identifier	Identifier string.
0x90	Large	Length of RO area	RO data	VPD-R list containing one or more VPD keywords This part is optional and might not appear.
0x91	Large	Length of R/W area	RW data	VPD-W list containing one or more VPD keywords. This part is optional and might not appear.
0x78	Small	N/A	N/A	End tag.

Note: The VPD-R and VPD-W structures can be in any order.

If the controller doesn't detect a value of 0x82 in the first byte of the VPD area, or the structure doesn't follow the description listed in Table 22-17, it assumes the area is not programmed and the entire 256 bytes area is read only. If a VPD-W tag is found after the VPD-R tag, the area defined by it's size is writable via the VPD structure. See the PCI 3.0 specification (Appendix I) for details of the different tags.

In any case, the VPD area is accessible for read and write via the regular EEPROM mechanisms.

The VPD area can be accessed through the PCIe* configuration space VPD capability structure. Write accesses to a read-only area or any access outside of the VPD area via this structure are ignored.

Note: Write access to Dwords, which are only partially in the read/write area, are ignored. It is responsibility of VPD software to make the right alignment to enable a write to the entire area.

22.4 Configurable I/O Pins

22.4.1 General-Purpose I/O (Software-Definable Pins)

The controller has two software-defined pins (SDP pins) per port that can be used for miscellaneous hardware or software-controllable purposes. These pins and their function are bound to a specific LAN device. For example, eight SDP pins cannot be associated with a single LAN device. These pins can each be individually configurable to act as either input or output pins. The default direction of each of the four pins is configurable via the EEPROM as well as the default value of any pins configured as outputs. To avoid signal contention, both pins are set as input pins until after the EEPROM configuration has been loaded.

In addition to both pins being individually configurable as inputs or outputs, they can be configured for use as General-Purpose Interrupt (GPI) inputs. To act as GPI pins, the desired pins must be configured as inputs. A separate GPI interrupt-detection enable is

then used to enable rising-edge detection of the input pin (rising-edge detection occurs by comparing values sampled at the internal clock rate as opposed to an edge-detection circuit). When detected, a corresponding GPI interrupt is indicated in the Interrupt Cause register.

The use, direction, and values of SDP pins are controlled and accessed using fields in the Device Control (CTRL) register and Extended Device Control (CTRL_EXT) register.

The SDPs can be used for special purpose mechanism such as watchdog indication (see Section 22.4.2 for details) or IEEE 1588 support.

22.4.2 Software Watchdog

In some situations it might be useful to give an indication to manageability firmware or to external devices that the controller hardware or the software device driver is not functional. For example, in a pass-through NIC, the controller might be bypassed if it is not functional. In order to provide this functionality, a watchdog mechanism is used. This mechanism can be enabled by default, according to EEPROM configuration. Once the host driver is up and it determines that hardware is functional, it might reset the watchdog timer to indicate that the controller is functional. The software device driver then should re-arm the timer periodically. If the timer is not re-armed after pre-programmed timeout (such as the timer runs out before it is reset), an interrupt is given to firmware and a pre-programmed SDP (either SDP0[0] or SDP1[0]) is raised. The SDP indication is shared between the ports.

The register controlling this feature is WDSTP. This register enables the setting of a time-out period and the activation of this mode. Both get their default from the EEPROM.

The re-arming of the timer is done by setting the *WDSWSTS.Dev_functional* bit.

If software needs to trigger the watchdog immediately because it suspects hardware is stuck, it can set the *WDSWSTS.Force_WD* bit. It can also give firmware an indication of the watchdog reason for using the *WDSWSTS.stuck_reason* field.

The SDP on which the watchdog indication is indicated, is set using the *CTRL.SDP0_WDE bit.* In this mode the *CTRL.SDP0_IODIR* should be set to output. The *CTRL.SDP0_DATA* bit indicates the polarity of the indication. Setting this bit in one of the cores causes the watchdog indications of all ports to be indicated on this SDP.

22.4.2.1 Watchdog Rearm

After a watchdog indication was received, in order to rearm the mechanism the following flow should be used:

1. Clear WD_enable bit in the WDSTP register.
2. Clear SDP0_WDE bit in CTRL register.
3. Set SDP0_WDE bit in CTRL register.
4. Set WD_enable in the WDSTP register.

22.4.3 LEDs

The controller provides one LED per port that can be used to indicate different statuses of the traffic. The default setup of the LEDs is done via EEPROM word offsets 0x1C and 0x1F from start of relevant LAN port section (LaN port 0, 1, 2 and 3). This setup is reflected in the LEDCTL register of each port. Each software device driver can change its setup individually. For each of the LEDs the following parameters can be defined:

- Mode: Defines which information is reflected by this LED. The encoding is described in the LEDCTL register.
- Polarity: Defines the polarity of the LED.
- Blink mode: Determines whether or not the LED should blink or be stable.

In addition, the blink rate of all LEDs can be defined. The possible rates are 200 ms or 83 ms for each phase. There is one rate for all the LEDs of a port.

22.5 Network Interfaces

22.5.1 Overview

The controller MAC provides a complete CSMA/CD function supporting IEEE 802.3 (10 Mb/s), 802.3u (100 Mb/s), 802.3z and 802.3ab (1000 Mb/s) implementations. The controller performs all of the functions required for transmission, reception, and collision handling called out in the standards.

Each controller MAC can be configured to use a different media interface. The controller supports the following potential configurations:

- External SerDes device such as an optical SerDes (SFP or on board) or backplane (1000BASE-BX or 1000BASE-KX) connections.
- External SGMII device. This mode is used for connections to external 10/100/1000 BASE-T PHYs that support the SGMII MAC interface.

Selection between the various configurations is programmable via each MAC's Extended Device Control register (*CTRL_EXT.LINK_MODE* bits) and default is set via EEPROM settings. Table 22-18 lists the encoding on the *LINK_MODE* field for each of the modes.

Table 22-18. Encoding on LINK_MODE Field for Each Mode

Link Mode	GBE Controller Mode
000b	reserved
001b	1000BASE-KX
010b	SGMII
011b	SerDes (1000BASE-BX)

The GMII/MII interface, used to communicate between the SGMII PCS, supports 10/100/1000 Mb/s operation, with both half- and full-duplex operation at 10/100 Mb/s, and only full-duplex operation at 1000 Mb/s.

The SerDes function can be used to implement a fiber-optics-based solution or backplane connection without requiring an external TBI mode transceiver/SerDes.

The SerDes interface can be used to connect to SFP modules. As such, this SerDes interface has the following limitations:

- No Tx clock
- AC coupling only

22.5.2 MAC Functionality

22.5.2.1 Internal GMII/MII Interface

The controller's MAC and PCS communicate through an internal MII interface that can be configured for 10/100 Mb/s (MII) mode of operation. For proper network operation, both the MAC and PHY must be properly configured (either explicitly via software or via hardware auto-negotiation) to identical speed and duplex settings.

All MAC configuration is performed using Device Control registers mapped into system memory or I/O space. In addition an external MDIO/MDC interface is available to configure external PHY's that are connected to the controller via the SGMII interface.

22.5.2.2 MDIO/MDC PHY Management Interface

The controller implements an IEEE 802.3 MII Management Interface (also known as the Management Data Input/Output or MDIO Interface) between the MAC and a PHY. This interface provides the MAC and software the ability to monitor and control the state of the PHY. The MDIO interface defines a physical connection, a special protocol that runs across the connection, and an internal set of addressable registers. The interface consists of a data line (MDIO) and clock line (MDC), which are accessible by software via the MAC register space.

- MDC (management data clock): This signal is used by the PHY as a clock timing reference for information transfer on the MDIO signal. The MDC is not required to be a continuous signal and can be frozen when no management data is transferred. The MDC signal has a maximum operating frequency of 2 MHz.
- MDIO (management data I/O): This used to transfer control and status information to and from the PHY (to read and write the PHY management registers).

Software can use MDIO accesses to read or write registers of an external SGMII PHY, by accessing MDIC register. MDIO configuration setup (External PHY, PHY Address and Shared MDIO) is defined in the MDICNFG register.

When working in SGMII/SerDes mode, the external PHY (if it exists) can be accessed either through MDC/MDIO as previously described, or via a two wire I^2C interface bus using the I2CCMD register (see Chapter 28.0, "GbE Programming Interface"). The two wire interface bus or the MDC/MDIO bus are connected via the same pins, and thus are mutually exclusive. In order to be able to control an external device, either by I^2C or MDC/MDIO, the 2 wires *SFP Enable* bit in *Initialization Control 3* EEPROM word, that's loaded into the *CTRL_EXT.I2C Enabled* register bit, should be set.

Each port has its own MDC/MDIO or two wire interface bus. However, the MDC/MDIO bus of LAN port 0 may be shared by all ports configured to external PHY operation (set to 1), to allow control of a multi PHY chip with a single MDC/MDIO bus.

MDIO operation using a shared bus or a separate bus is controlled by the *MDICNFG.Com_MDIO* bit that's loaded from *Initialization Control 3* EEPROM word following reset. The external port PHY Address is written in the *MDICNFG.PHYADD* register field and is loaded from the *Initialization Control 4* EEPROM word following reset.

22.5.2.2.1 Detection of External I^2C or MDIO Connection

When the *CTRL_EXT.I2C Enabled* bit is set to 1, Software can recognize type of external PHY control bus (MDIO or I^2C) connection according to the values loaded from the EEPROM to the *MDICNFG.Destination* bit and the *CTRL_EXT.LINK_MODE* field in the following manner:

- External I^2C operating mode - *MDICNFG.Destination* equals 0 and *CTRL_EXT.LINK_MODE* is not equal to 0.
- External MDIO Operating mode - *MDICNFG.Destination* equals 1 and *CTRL_EXT.LINK_MODE* is not equal to 0.

22.5.2.2.2 MDIC and MDICNFG Register Usage

Note: MDIC requires ownership acquisition of Shared Resources. See Section 23.5.1, "Acquiring Ownership Over a Shared Resource" on page 1057 for shared resource acquisition flow.

The following read/write cycles should be executed only after ownership of shared resource has be acquired.

For a MDIO read cycle, the sequence of events is as follows:

1. If default MDICNFG register values loaded from EEPROM need to be updated. The processor performs a PCIe* write access to the *MDICNFG* register to define the:
   — PHYADD = Address of external PHY.
   — Destination = External PHY.
   — Com_MDIO = Shared or separate MDIO external PHY connection.
2. The processor performs a PCIe* write cycle to the MDIC register with:
   — Ready = 0b
   — Interrupt Enable set to 1b or 0b
   — Opcode = 10b (read)
   — REGADD = Register address of the specific register to be accessed (0 through 31).
3. The MAC applies the following sequence on the MDIO signal to the PHY:

   <PREAMBLE><01><10><PHYADD><REGADD><Z> where Z stands for the MAC tri-stating the MDIO signal.
4. The PHY returns the following sequence on the MDIO signal:

   <0><DATA><IDLE>.
5. The MAC discards the leading bit and places the following 16 data bits in the MII register.
6. The controller asserts an interrupt indicating MDIO "Done" if the *Interrupt Enable* bit was set.
7. The controller sets the *Ready* bit in the MDIC register indicating the Read is complete.
8. The processor might read the data from the MDIC register and issue a new MDIO command.

For a MDIO write cycle, the sequence of events is as follows:

1. If default MDICNFG register values loaded from EEPROM need to be updated. The processor performs a PCIe* write cycle to the MDICNFG register to define the:
   — PHYADD = Address of external PHY.
   — Destination = External PHY.
   — Com_MDIO = Shared or separate MDIO external PHY connection.

2. The processor performs a PCIe* write cycle to the MDIC register with:
   — Ready = 0b.
   — Interrupt Enable set to 1b or 0b.
   — Opcode = 01b (write).
   — REGADD = Register address of the register to be accessed (0 through 31).
   — Data = Specific data for desired control of the PHY.
3. The MAC applies the following sequence on the MDIO signal to the PHY:

   <PREAMBLE><01><01><PHYADD><REGADD><10><DATA><IDLE>
4. The controller asserts an interrupt indicating MDIO "Done" if the *Interrupt Enable* bit was set.
5. The controller sets the *Ready* bit in the MDIC register to indicate that the write operation completed.
6. The CPU might issue a new MDIO command.

Note: A MDIO read or write might take as long as 64 μs from the processor write to the *Ready* bit assertion. When a shared MDC/MDIO bus is used, each transaction can take up to 256 μs to complete if other ports are using the bus concurrently.

If an invalid opcode is written by software, the MAC does not execute any accesses to the PHY registers.

If the PHY does not generate a 0b as the second bit of the turn-around cycle for reads, the MAC aborts the access, sets the *E* (error) bit, writes 0xFFFF to the data field to indicate an error condition, and sets the *Ready* bit.

Note: After a PHY reset, access through the MDIC register should not be attempted for 300 μsec.

22.5.3 SerDes, SGMII and 1000BASE-KX Support

The controller can be configured to follow either SGMII, Serial-SerDes or 1000BASE-KX standards. When in SGMII mode, the controller can be configured to operate in 1 Gb/s, 100 Mb/s or 10 Mb/s speeds. When in the 10/100 Mb/s speed, the controller can be configured to half-duplex mode of operation. When configured for Serial-SerDes or 1000BASE-KX operation, the port supports only 1 Gb/s, full-duplex operation. Since the serial interfaces are defined as differential signals, internally the hardware has analog and digital blocks. Following is the initialization/configuration sequence for the analog and digital blocks.

22.5.3.1 SerDes, SGMII and 1000BASE-KX PCS Block

The link setup for SerDes, 1000BASE-KX and SGMII are described in sections 22.5.4.1, 22.5.4.2 and 22.5.4.3 respectively.

22.5.3.2 GbE Physical Coding Sub-Layer (PCS)

The controller integrates the 802.3z PCS function on-chip.

The packet encapsulation is based on the Fiber Channel (FC0/FC1) physical layer and uses the same coding scheme to maintain transition density and DC balance. The physical layer device is the SerDes and is used for 1000BASE-SX, -L-, or -CX configurations.

22.5.3.2.1 8B10B Encoding/Decoding

The GbE PCS circuitry uses the same transmission-coding scheme used in the fiber channel physical layer specification. The 8B10B-coding scheme was chosen by the standards committee in order to provide a balanced, continuous stream with sufficient transition density to allow for clock recovery at the receiving station. There is a 25% overhead for this transmission code, which accounts for the data-signaling rate of 1250 Mb/s with 1000 Mb/s of actual data.

22.5.3.2.2 Code Groups and Ordered Sets

Code group and ordered set definitions are defined in clause 36 of the IEEE 802.3z standard. These represent special symbols used in the encapsulation of GbE packets. The following table contains a brief description of defined ordered sets and included for informational purposes only. See clause 36 of the IEEE 802.3z specification for more details.

Table 22-19. Ordered Sets

Code	Ordered_Set	# of Code Groups	Usage
/C/	Configuration	4	General reference to configuration ordered sets, either /C1/ or /C2/, which is used during auto-negotiation to advertise and negotiate link operation information between link partners. Last 2 code groups contain configuration base and next page registers.
/C1/	Configuration 1	4	See /C/. Differs from /C2/ in 2nd code group for maintaining proper signaling disparity[1].
/C2/	Configuration 2	4	See /C/. Differs from /C1/ in 2nd code group for maintaining proper signaling disparity[1].
/I/	IDLE	2	General reference to idle ordered sets. Idle characters are continually transmitted by the end stations and are replaced by encapsulated packet data. The transitions in the idle stream enable the SerDes to maintain clock and symbol synchronization between link partners.
/I1/	IDLE 1	2	See /I/. Differs from /I2/ in 2nd code group for maintaining proper signaling disparity[1].
/I2/	IDLE 2	2	See /I/. Differs from /I1/ in 2nd code group for maintaining proper signaling disparity[1].
/R/	Carrier_Extend	1	This ordered set is used to indicate carrier extension to the receiving PCS. It is also used as part of the end_of_packet encapsulation delimiter as well as IPG for packets in a burst of packets.
/S/	Start_of_Packet	1	The SPD (start_of_packet delimiter) ordered set is used to indicate the starting boundary of a packet transmission. This symbol replaces the last byte of the preamble received from the MAC layer.
/T/	End_of_Packet	1	The EPD (end_of_packet delimiter) is comprised of three ordered sets. The /T/ symbol is always the first of these and indicates the ending boundary of a packet.
/V/	Error_Propagation	1	The /V/ ordered set is used by the PCS to indicate error propagation between stations. This is normally intended to be used by repeaters to indicate collisions.

1. The concept of running disparity is defined in the standard. In summary, this refers to the 1-0 and 0-1 transitions within 8B10B code groups.

22.5.4 Auto-Negotiation and Link Setup Features

The method for configuring the link between two link partners is highly dependent on the mode of operation as well as the functionality provided by the specific physical layer device (SerDes). In SerDes mode, the PCH provides the complete PCS and Auto-negotiation functionality as defined in IEEE802.3 clause 36 and clause 37.In SGMII mode, the PCH supports the SGMII link auto-negotiation process, whereas the link auto-negotiation, as defined in is done by the external PHY. In 1000BASE-KX

mode, the PCH supports only parallel detect of 1000BASE-KX signaling and does not support the full Auto-Negotiation for Backplane Ethernet protocol as defined in IEEE802.3ap clause 73.

Configuring the link can be accomplished by several methods ranging from software forcing link settings, software-controlled negotiation, MAC-controlled auto-negotiation, to auto-negotiation initiated by a PHY. The following sections describe processes of bringing the link up including configuration of the PCH and the transceiver, as well as the various methods of determining duplex and speed configuration.

The process of determining link configuration differs slightly based on the specific link mode (external SerDes, SGMII or 1000BASE-KX) being used.

When operating in a SerDes mode, the PCS layer performs auto-negotiation per clause 37 of the 802.3z standard. The transceiver used in this mode does not participate in the auto-negotiation process as all aspects of auto-negotiation are controlled by the PCH.

When operating in SGMII mode, the PCS layer performs SGMII auto-negotiation per the SGMII specification. The external PHY is responsible for the Ethernet auto-negotiation process.

When operating in 1000BASE-KX mode the PCH performs parallel detect of 1000BASE-KX operation but does not implement the full Auto-Negotiation for Backplane Ethernet sequence as defined in IEEE802.3ap clause 73.

22.5.4.1 SerDes Link Configuration

When using SerDes link mode, link mode configuration can be performed using the PCS function in the PCH. The hardware supports both hardware and software auto-negotiation methods for determining the link configuration, as well as allowing for a manual configuration to force the link. Hardware auto-negotiation is the preferred method.

22.5.4.1.1 Signal Detect Indication

The SRDS_0/1/2/3_SIG_DET pins can be connected to a Signal Detect or loss-of-signal (LOS) output that indicates when no laser light is being received when the PCH is used in a 1000BASE-SX or -LX implementation (SerDes operation). It prevents false carrier cases occurring when transmission by a non connected port couples in to the input. Unfortunately, there is no standard polarity for this signal coming from different manufacturers. The *CTRL.ILOS* bit provides for inversion of the signal from different external optical module vendors, and should be set when the external optical module provides a negative-true loss-of-signal.

22.5.4.1.2 MAC Link Speed

SerDes operation is only defined for 1000 Mb/s operation. Other link speeds are not supported. When configured for the SerDes interface, the MAC speed-determination function is disabled and the Device Status register bits (STATUS.SPEED) indicate a value of 10b for 1000 Mb/s.

22.5.4.1.3 SerDes Mode Auto-Negotiation

In SerDes mode, after power up or reset via PCIE_EP_RST#, the controller initiates IEEE802.3 clause 37 auto-negotiation based on the default settings in the device control and transmit configuration or PCS Link Control Word registers, as well as settings read from the EEPROM. If enabled in the EEPROM, the PCH immediately performs auto-negotiation.

TBI mode auto-negotiation, as defined in clause 37 of the IEEE 802.3z standard, provides a protocol for two devices to advertise and negotiate a common operational mode across a GbE link. The controller fully supports the IEEE 802.3z auto-negotiation function when using the on-chip PCS and internal SerDes.

TBI mode auto-negotiation is used to determine the following information:

- Duplex resolution (even though the PCH MAC only supports full-duplex in SerDes mode).
- Flow control configuration.

Note: Since speed for SerDes modes is fixed at 1000 Mb/s, speed settings in the Device Control register are unaffected by the auto-negotiation process.

- Auto-negotiation can be initiated at power up or by asserting PCIE_EP_RST# and enabling specific bits in the EEPROM.

The auto-negotiation process is accomplished by the exchange of /C/ ordered sets that contain the capabilities defined in the PCS_ANADV register in the 3rd and 4th symbols of the ordered sets. Next page are supported using the PCS_NPTX_AN register.

Bits *FD* and *LU* in the Device Status (STATUS) register, and bits in the PCS_LSTS register provide status information regarding the negotiated link.

Auto-negotiation can be initiated by the following:

- PCS_LCMD.AN_ENABLE transition from 0b to 1b
- Receipt of /C/ ordered set during normal operation
- Receipt of a different value of the /C/ ordered set during the negotiation process
- Transition from loss of synchronization to synchronized state (if AN_ENABLE is set).
- PCS_LCMD.AN_RESTART transition from 0b to 1b

Resolution of the negotiated link determines device operation with respect to flow control capability and duplex settings. These negotiated capabilities override advertised and software-controlled device configuration.

Software must configure the *PCS_ANADV* fields to the desired advertised base page. The bits in the Device Control register are not mapped to the *txConfigWord* field in hardware until after auto-negotiation completes. Table 22-20 lists the mapping of the *PCS_ANADV* fields to the Config_reg Base Page encoding per clause 37 of the standard.

Table 22-20. 802.3z Advertised Base Page Mapping

15	14	13:12	11:9	8:7	6	5	4:0
Nextp	Ack	RFLT	rsv	ASM	Hd	Fd	rsv

The partner advertisement can be seen in the PCS_ LPAB and PCS_ LPABNP registers.

22.5.4.1.4 Forcing Link-up in SerDes Mode

Forcing link can be accomplished by software by writing a 1b to *CTRL.SLU*, which forces the MAC PCS logic into a link-up state (enables listening to incoming characters when SRDS_[n]_SIG_DET is asserted by the external optical module.

Note: The *PCS_LCMD.AN_ENABLE* bit must be set to a logic zero to enable forcing link.

- When link is forced via the *CTRL.SLU* bit, the link does not come up unless the SRDS_[n]_SIG_DET signal is asserted or an internal energy indication is received from the SerDes receiver, implying that there is a valid signal being received by the optical module or SerDes circuitry.

The source of the signal detect is defined by the *ENRGSRC* bit in the CONNSW register.

22.5.4.1.5 HW Detection of Non-Auto-Negotiation Partner

Hardware can detect a SerDes link partner that sends idle code groups continuously, but does not initiate or answer an auto-negotiation process. In this case, hardware initiates an auto-negotiation process, and if it fails after some timeout, a link up is assumed. To enable this functionality the *PCS_LCTL.AN_TIMEOUT_EN* bit should be set. This mode can be used instead of the force link mode as a way to support a partner that do not support auto-negotiation.

22.5.4.2 1000BASE-KX Link Configuration

When using 1000BASE-KX link mode, link mode configuration is forced manually by software since the PCH does not support IEEE802.3 clause 73 backplane auto-negotiation.

22.5.4.2.1 MAC Link Speed

1000BASE-KX operation is only defined for 1000 Mb/s operation. Other link speeds are not supported. When configured for the 1000BASE-KX interface, the MAC speed-determination function is disabled and the Device Status register bits (STATUS.SPEED) indicate a value of 10b for 1000 Mb/s.

22.5.4.2.2 1000BASE-KX Auto-Negotiation

The controller only supports parallel detection of the 1000BASE-KX link and does not support the full IEEE802.3ap clause 73 backplane auto-negotiation protocol.

22.5.4.2.3 Forcing Link-up in 1000BASE-KX Mode

In 1000BASE-KX mode (*EXT_CTRL.LINK_MODE* = 001b) the controller always operates in force link mode. The MAC PCS logic is placed in a link-up state once energy indication, implying that a valid signal is being received by the 1000BASE-KX circuitry, is received. When in the link-up state PCS logic can lock on incoming characters.

Note: The *PCS_LCMD.AN_ENABLE* bit has no effect when in 1000BASE-KX mode. Auto-negotiation is always disabled on ports operating in 1000BASE-KX mode. The source of the signal detect in 1000BASE-KX is internal and the CONNSW.*ENRGSRC* bit has no effect. In 1000BASE-KX mode the PCH always operates in force link mode and the value of the *CTRL.SLU* bit has no effect on link-up detection

22.5.4.2.4 1000BASE-KX HW Detection of Link Partner

In 1000BASE-KX mode, hardware detects a 1000BASE-KX link partner that sends idle or none idle code groups continuously. In 1000BASE-KX operation force link-up mode is used.

22.5.4.3 SGMII Link Configuration

When working in SGMII mode, the actual link setting is done by the external PHY and is dependent on the settings of this PHY. The SGMII auto-negotiation process described in the sections that follow is only used to establish the MAC/PHY connection.

22.5.4.3.1 SGMII Auto-Negotiation

This auto-negotiation process is not dependent on the SRDS_[n]_SIG_DET signal, as this signal indicates optical module signal detection and is not relevant in SGMII mode.

The outcome of this auto-negotiation process includes the following information:

- Link status
- Speed
- Duplex

This information is used by hardware to configure the MAC, when operating in SGMII mode.

Bits *FD* and *LU* of the Device Status (STATUS) register and bits in the PCS_LSTS register provide status information regarding the negotiated link.

Auto-negotiation can be initiated by the following:

- PCS_LCMD.AN_ENABLE transition from 0b to 1b.
- Receipt of /C/ ordered set during normal operation.
- Receipt of different value of the /C/ ordered set during the negotiation process.
- Transition from loss of synchronization to a synchronized state (if AN_ENABLE is set).
- PCS_LCMD.AN_RESTART transition from 0b to 1b.

Auto-negotiation determines the controller operation with respect to speed and duplex settings. These negotiated capabilities override advertised and software controlled device configuration.

When working in SGMII mode, there is no need to set the PCAS_ANADV register, as the MAC advertisement word is fixed. The result of the SGMII level auto-negotiation can be read from the PCS_LPAB register.

22.5.4.3.2 Forcing Link in SGMII Mode

In SGMII, forcing of the link cannot be done at the PCS level, only in the external PHY. The forced speed and duplex settings are reflected by the SGMII auto-negotiation process; the MAC settings are automatically done according to this functionality.

22.5.4.3.3 MAC Speed Resolution

The MAC speed and duplex settings are always set according to the SGMII auto-negotiation process.

22.5.4.4 Loss of Signal/Link Status Indication

For all modes of operation, an LOS/LINK signal provides an indication of physical link status to the MAC. When the MAC is configured for optical SerDes mode, the input reflects loss-of-signal connection from the optics. In backplane mode, where there is no LOS external indication, an internal indication from the SerDes receiver can be used. In

SFP systems the LOS indication from the SFP can be used. Assuming that the MAC has been configured with *CTRL.SLU*=1b, the MAC status bit *STATUS.LU*, when read, generally reflects whether SerDes has link.

When loss-of-signal asserted from the SerDes is asserted, the MAC considers this to be a transition to a link-down situation (such as cable unplugged, loss of link partner, etc.). If the Link Status Change (LSC) interrupt is enabled, the MAC generates an interrupt to be serviced by the software device driver.

22.5.5 Ethernet Flow Control (FC)

The controller supports flow control as defined in 802.3x as well as the specific operation of asymmetrical flow control defined by 802.3z.

Flow control is implemented as a means of reducing the possibility of receive buffer overflows, which result in the dropping of received packets, and allows for local controlling of network congestion levels. This can be accomplished by sending an indication to a transmitting station of a nearly full receive buffer condition at a receiving station.

The implementation of asymmetric flow control allows for one link partner to send flow control packets while being allowed to ignore their reception. For example, not required to respond to PAUSE frames.

The following registers are defined for the implementation of flow control:

- *CTRL.RFCE* field is used to enable reception of legacy flow control packets and reaction to them.
- *CTRL.TFCE* field is used to enable transmission of legacy flow control packets.
- Flow Control Address Low, High (FCAL/H) - 6-byte flow control multicast address
- Flow Control Type (FCT) 16-bit field to indicate flow control type
- Flow Control bits in Device Control (CTRL) register - Enables flow control modes.
- Discard PAUSE Frames (DPF) and Pass MAC Control Frames (PMCF) in RCTL - controls the forwarding of control packets to the host.
- Flow Control Receive Threshold High (FCRTH) - A 13-bit high watermark indicating receive buffer fullness. A single watermark is used in link FC mode.
- Flow Control Receive Threshold Low (FCRTL) - A 13-bit low watermark indicating receive buffer emptiness. A single watermark is used in link FC mode.
- Flow Control Transmit Timer Value (FCTTV) - a set of 16-bit timer values to include in transmitted PAUSE frame. A single timer is used in Link FC mode.
- Flow Control Refresh Threshold Value (FCRTV) - 16-bit PAUSE refresh threshold value

22.5.5.1 MAC Control Frames and Receiving Flow Control Packets

22.5.5.1.1 Structure of 802.3X FC Packets

Three comparisons are used to determine the validity of a flow control frame:

1. A match on the 6-byte multicast address for MAC control frames or to the station address of the controller (Receive Address Register 0).
2. A match on the type field.
3. A comparison of the MAC *Control Op-Code* field.

The 802.3x standard defines the MAC control frame multicast address as 01-80-C2-00-00-01.

The *Type* field in the FC packet is compared against an IEEE reserved value of 0x8808.

The final check for a valid PAUSE frame is the MAC control op-code. At this time only the PAUSE control frame op-code is defined. It has a value of 0x0001.

Frame-based flow control differentiates XOFF from XON based on the value of the *PAUSE* timer field. Non-zero values constitute XOFF frames while a value of zero constitutes an XON frame. Values in the *Timer* field are in units of pause quantum (slot time). A pause quantum lasts 64 byte times, which is converted in absolute time duration according to the line speed.

Note: XON frame signals the cancellation of the pause from initiated by an XOFF frame - pause for zero pause quantum.

Table 22-21 lists the structure of a 802.3X FC packet.

Table 22-21. 802.3X Packet Format

DA	01_80_C2_00_00_01 (6 bytes)
SA	Port MAC address (6 bytes)
Type	0x8808 (2 bytes)
Op-code	0x0001 (2 bytes)
Time	XXXX (2 bytes)
Pad	42 bytes
CRC	4 bytes

22.5.5.1.2 Operation and Rules

The controller operates in Link FC.

- Link FC is enabled by the RFCE bit in the CTRL Register.

Note: Link flow control capability is negotiated between link partners via the auto negotiation process. It is the software device driver responsibility to reconfigure the link flow control configuration after the capabilities to be used where negotiated as it might modify the value of these bits based on the resolved capability between the local device and the link partner.

Once the receiver has validated receiving an XOFF, or PAUSE frame, the PCH performs the following:

- Increments the appropriate statistics register(s)
- Sets the *Flow_Control State* bit in the FCSTS0 register.

- Initializes the pause timer based on the packet's *PAUSE* timer field (overwriting any current timer's value)
- Disables packet transmission or schedules the disabling of transmission after the current packet completes.

Resumption of transmission might occur under the following conditions:

- Expiration of the PAUSE timer
- Reception of an XON frame (a frame with its PAUSE timer set to 0b)

Both conditions clear the relevant *Flow_Control State* bit in the relevant FCSTS0 register and transmission can resume. Hardware records the number of received XON frames.

22.5.5.1.3 Timing Considerations

When operating at 1 Gb/s line speed, the controller must not begin to transmit a (new) frame more than two pause-quantum-bit times after receiving a valid link XOFF frame, as measured at the wires. A pause quantum is 512-bit times.

When operating in full duplex at 100 Mb/s speeds, the controller must not begin to transmit a (new) frame more than 576-bit times after receiving a valid link XOFF frame, as measured at the wire.

22.5.5.2 PAUSE and MAC Control Frames Forwarding

Two bits in the Receive Control register, control forwarding of PAUSE and MAC control frames to the host. These bits are *Discard PAUSE Frames* (*DPF*) and *Pass MAC Control Frames* (*PMCF*):

- The *DPF* bit controls forwarding of PAUSE packets to the host.
- The *PMCF* bit controls forwarding of non-PAUSE packets to the host.

Note: When flow control reception is disabled (CTRL.RFCE = 0), legacy flow control packets are not recognized and are parsed as regular packets.

Table 22-22 lists the behavior of the *DPF* bit.

Table 22-22. Forwarding of PAUSE Packet to Host (DPF Bit)

RFCE	DPF	Are FC Packets Forwarded to Host?
0	X	Yes. Packets needs to pass the L2 filters (see).[1]
1	0	Yes. Packets needs to pass the L2 filters

1. The flow control multicast address is not part of the L2 filtering unless explicitly required.

22.5.5.3 Transmission of PAUSE Frames

The controller generates PAUSE packets to ensure there is enough space in its receive packet buffers to avoid packet drop. The controller monitors the fullness of its receive packet buffers and compares it with the contents of a programmable threshold. When the threshold is reached, the controller sends a PAUSE frame. The controller supports the sending of link Flow Control (FC).

Note: Similar to receiving link flow control packets previously mentioned, link XOFF packets can be transmitted only if this configuration has been negotiated between the link

partners via the auto-negotiation process or some higher level protocol. The setting of this bit by the software device driver indicates the desired configuration.

Note: The transmission of flow control frames should only be enabled in full-duplex mode per the IEEE 802.3 standard. Software should ensure that the transmission of flow control packets is disabled when the PCH is operating in half-duplex mode.

22.5.5.3.1 Operation and Rules

Transmission of link PAUSE frames is enabled by software writing a 1b to the *TFCE* bit in the Device Control register.

The content of the Flow Control Receive Threshold High (FCRTH) register determines at what point the controller first transmits a PAUSE frame. The controller monitors the fullness of the receive packet buffer and compares it with the contents of FCRTH. When the threshold is reached, the PCH sends a PAUSE frame with its pause time field equal to FCTTV.

At this time, the controller starts counting an internal shadow counter (reflecting the pause timeout counter at the partner end) from zero. When the counter reaches the value indicated in FCRTV register, then, if the PAUSE condition is still valid (meaning that the buffer fullness is still above the high watermark), a XOFF message is sent again.

Once the receive buffer fullness reaches the low water mark, the controller sends a XON message (a PAUSE frame with a timer value of zero). Software enables this capability with the *XONE* field of the FCRTL.

The controller sends an additional PAUSE frame if it has previously sent one and the packet buffer overflows. This is intended to minimize the amount of packets dropped if the first PAUSE frame did not reach its target.

22.5.5.3.2 Software Initiated PAUSE Frame Transmission

The controller has the added capability to transmit an XOFF frame via software. This is accomplished by software writing a 1b to the *SWXOFF* bit of the Transmit Control register. Once this bit is set, hardware initiates the transmission of a PAUSE frame in a manner similar to that automatically generated by hardware.

The *SWXOFF* bit is self-clearing after the PAUSE frame has been transmitted.

Note: The Flow Control Refresh Threshold mechanism does not work in the case of software-initiated flow control. Therefore, it is the software's responsibility to re-generate PAUSE frames before expiration of the pause counter at the other partner's end.

The state of the *CTRL.TFCE* bit or the negotiated flow control configuration does not affect software generated PAUSE frame transmission.

Note: Software sends an XON frame by programming a 0b in the PAUSE timer field of the *FCTTV* register. Software generation of XON packet is not allowed while the hardware flow control mechanism is active, as both use the *FCTTV* registers for different purposes.

- XOFF transmission is not supported in 802.3x for half-duplex links. Software should not initiate an XOFF or XON transmission if the PCH is configured for half-duplex operation.
- When flow control is disabled, pause packets (XON, XOFF, and other FC) are not detected as flow control packets and can be counted in a variety of counters (such as multicast).

22.5.5.4 IPG Control and Pacing

The controller supports the following modes of controlling IPG duration:

- Fixed IPG - the IPG is extended by a fixed duration

22.5.5.4.1 Fixed IPG Extension

The controller allows controlling of the IPG duration. The IPGT configuration field enables an extension of IPG in 4-byte increments. One possible use of this capability is to allow the insertion of bytes into the transmit packet after it has been transmitted by the controller without violating the minimum IPG requirements. For example, a security device connected in series to the controller might add security headers to transmit packets before the packets are transmitted on the network.

22.5.6 Loopback Support

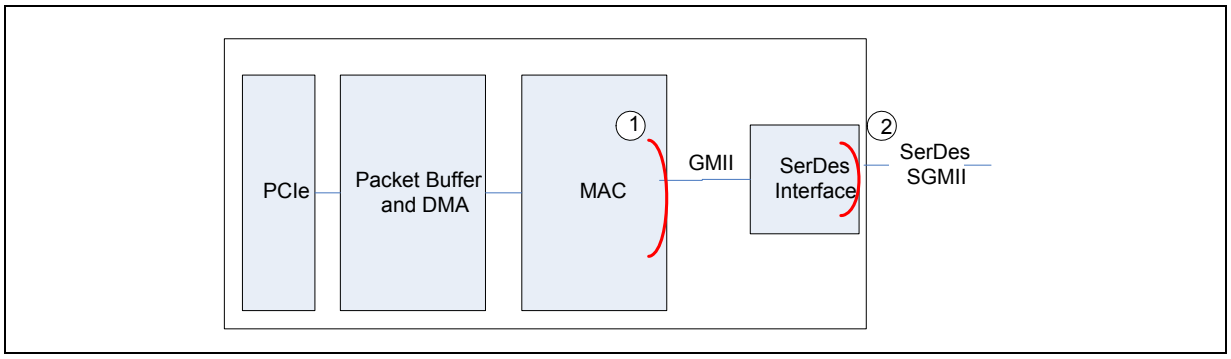
22.5.6.1 General

The controller supports the following types of internal loopback in the LAN interfaces:

- MAC Loopback (Point 1)
- SerDes, SGMII or 1000BASE-KX Loopback (Point 2)

By setting the device to loopback mode, packets that are transmitted towards the line will be looped back to the host. The controller is fully functional in these modes, just not transmitting data over the lines. The following figure shows the points of loopback.

Figure 22-9. GbE Controller Loopback Modes



22.5.6.2 MAC Loopback

In MAC loopback, the PHY and SerDes blocks are not functional and data is looped back before these blocks.

22.5.6.2.1 Setting GbE Controller to MAC Loopback Mode

The following procedure puts the controller in MAC loopback mode:

- Set *RCTL.LBM* to 2'b01 (bits 7:6)
- Set *CTRL.SLU* (bit 6, should be set by default)
- Set *CTRL.FRCSPD* & *FRCDPLX* (bits 11&12)
- Set *CTRL.SPEED* to 2'b10 (1G) and *CTRL.FD*

- Set *CTRL.ILOS*
- Set *CTRL_EXT.LINK_MODE* = 011b

Filter configuration and other TX/RX processes are the same as in normal mode.

22.5.6.3 SerDes, SGMII and 1000BASE-KX Loopback

In SerDes, SGMII, or 1000BASE-KX loopback mode, data is looped back at the end of the relevant functionality. All of the design that is functional in SerDes/SGMII or 1000BASE-KX mode is involved in the loopback.

Note: SerDes loopback is functional only if the SerDes link is up.

22.5.6.3.1 Setting GbE Controller to SerDes, SGMII, or 1000BASE-KX Loopback Mode

The following procedure places the GbE controller in SerDes loopback mode:

- Set MPHY to support SerDes Near End Digital Loopback:
  - GBE_MPHY_ADDR_CTRL[15:0] (CSR 0x0024 BITS 15:0) = 16'h0004
  - GBE_MPHY_DATA.DATA (CSR 0x0E10 BITS 31:0) = 32'h0000_0070
  - GBE_MPHY_ADDR_CTRL[15:0] (CSR 0x0024 BITS 15:0) = 16'h0008
  - GBE_MPHY_DATA.DATA (CSR 0x0E10 BITS 31:0) = 32'h0000_0800
  - GBE_MPHY_CTRL.MISC_CTRL (CSR 0x0E08 BITS 30:28) = 3'b010
  - MANC.RCV_TCO_EN (CSR 0x5820 BIT 17) = 0
  - RCTL.LBM (CSR 0x0100 BITS 7:6) = 3
- Set Link mode to either SerDes, SGMII or 1000BASE-KX by:
  - 1000BASE-KX: *CTRL_EXT.LINK_MODE* (CSR 0x18 BITS 24:22) = 001b
  - SGMII: *CTRL_EXT.LINK_MODE* (CSR 0x18 BITS 24:22) = 010b
  - SerDes: *CTRL_EXT.LINK_MODE* (CSR 0x18 BITS 24:22) = 011b
- Configure SERDES to loopback: *RCTL.LBM* = 11b
- Move to Force mode by setting the following bits:
  - *CTRL.FD* (CSR 0x0 bit 0) = 1
  - *CTRL.SLU* (CSR 0x0 bit 6) = 1
  - *CTRL.RFCE* (CSR 0x0 bit 27) = 0
  - *CTRL.TFCE* (CSR 0x0 bit 28) = 0
  - *PCS_LCTL.FORCE_LINK* (CSR 0X4208 bit 5) = 1
  - *PCS_LCTL.FSD* (CSR 0X4208 bit 4) = 1
  - *PCS_LCTL.FDV* (CSR 0X4208 bit 3) = 1
  - *PCS_LCTL.FLV* (CSR 0X4208 bit 0) = 1
  - *PCS_LCTL.AN_ENABLE* (CSR 0X4208 bit 16) = 0
  - *CONNSW.ENRGSRC* (CSR 0X0034 bit 2) = 0

§ §

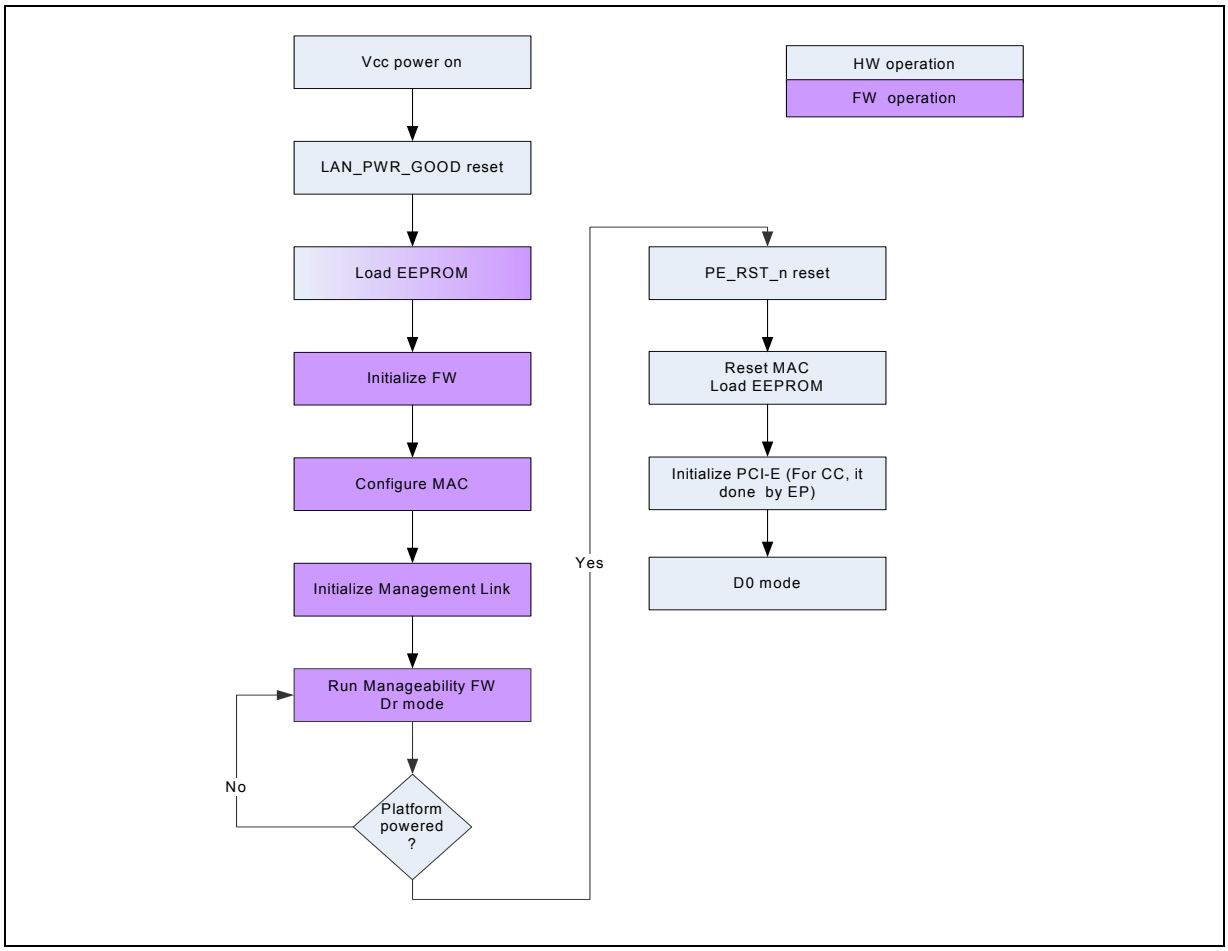
23.0 GbE Initialization

23.1 Power Up

23.1.1 Power-Up Sequence

The following figure shows the power-up sequence from power ramp up and to when the GbE Controller is ready to accept host commands.

Figure 23-1. Power-Up - General Flow



23.1.2 Power-Up Timing Diagram

Figure 23-2. Power-Up Timing Diagram

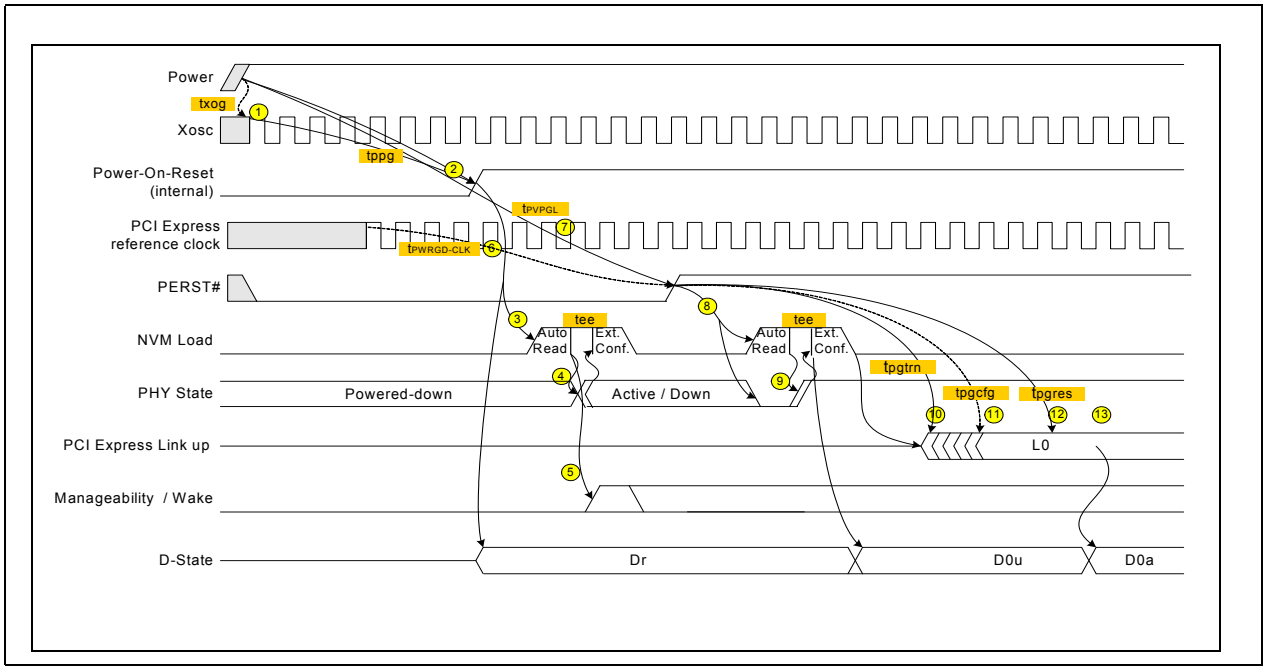


Table 23-1. Notes to Power-Up Timing Diagram

Note	Description
1	Xosc is stable t_{xog} after the power is stable
2	Internal Reset is released after all power supplies are good and t_{ppg} after Xosc is stable.
3	An NVM read starts on the rising edge of the internal Reset or GBE_AUX_PWR_OK.
4	After reading the NVM, PHY might exit power down mode.
5	APM Wakeup and/or manageability might be enabled based on NVM contents.
6	The PCIe* reference clock is valid $t_{PE\_RST-CLK}$ before the de-assertion of PCIE_EP_RST# (according to PCIe* spec).
7	PCIE_EP_RST# is de-asserted t_{PVPGL} after power is stable (according to PCIe* spec).
8	De-assertion of PCIE_EP_RST# causes the NVM to be re-read, and disables Wake Up.
9	After reading the NVM, PHY exits power-down mode.
10	Link training starts after t_{pgtrn} from PCIE_EP_RST# de-assertion.
11	A first PCIe* configuration access might arrive after t_{pgcfg} from PCIE_EP_RST# de-assertion.
12	A first PCI configuration response can be sent after tpgres from PCIE_EP_RST# de-assertion
13	Writing a 1 to the *Memory Access Enable* bit in the PCI *Command Register* transitions the device from D0u to D0 state.

23.2 Reset Operation

23.2.1 Hardware-Based Reset Sources

The Controller reset sources are described below.

23.2.1.1 GBE_AUX_PWR_OK

This reset comes directly from the GbE Cluster GBE_AUX_PWR_OK pin.

This reset will cause all sticky and non-sticky registers to be cleared and FSM/queues to reset to their initial value. This reset acts as a master reset of the entire Controller function. It is level sensitive, and while it is zero holds all of the registers in reset.

23.2.1.2 PCIE_EP_RST#

The de-assertion of PCIE_EP_RST# indicates that both the power and the PCIe* clock sources are stable. This pin asserts an internal reset also after a D3cold exit. Most units are reset on the rising edge of PCIE_EP_RST#.

23.2.1.3 Function Level Reset (FLR)

A FLR reset to a function is issued, by setting bit *15* in the *Device Control* configuration register, is equivalent to a D0 ⇨ D3 ⇨ D0 transition. The only difference is that this reset does not require driver intervention in order to stop the master transactions of this function. This reset is per function and resets only the function without effecting activity of other functions or Lan ports.

The EEPROM is partially reloaded after an FLR reset. The words read from EEPROM at FLR are the same as read following a full software reset. A list of these words can be found in Chapter 22.0, “GbE Interconnects”.

A FLR reset to a function resets all the queues, interrupts, and statistics registers attached to the function. It also resets PCIe* R/W configuration bits as well as disables transmit and receive flows for the queues allocated to the function. All pending read requests are dropped and PCIe* read completions to the function might be completed as unexpected completions and silently discarded (following update of flow control credits) without logging or signaling as an error.

Note: If software initiates a FLR when the *Transactions Pending* bit in the *Device Status* configuration register is set to 1b, then software must not initialize the function until allowing time for any associated completions to arrive. The *Transactions Pending* bit is cleared upon completion of the FLR.

23.3 Software Based Reset Sources

23.3.1 Full Software Reset (DEV_RST and RST)

Software can reset the Controller by either setting the Port Software Reset (*CTRL.RST)* or the Device Reset bit (*CTRL.DEV_RST*) in the Device Control Register. The Port Software Reset (*CTRL.RST*) is per function and resets only the function that received the software reset. The Device Reset (*CTRL.DEV_RST)* resets all functions and common logic. In both cases, PCI configuration space (configuration and mapping) of the device is unaffected. Prior to issuing software reset the driver needs to operate the Bus Master Enable algorithm as defined in Section 23.3.2.2.

Port Software Reset (*CTRL.RST*) can be used to reset the respective port. The RST bit is provided primarily to recover from an indeterminate or suspected Port hung hardware state. Most registers (receive, transmit, interrupt, statistics, etc.) and state machines in the port are set to their power-on reset values, approximating the state following a power-on or PCIe* reset. However, PCIe* configuration registers and logic common to all ports is not reset, leaving the device mapped into system memory space and accessible by a driver.

Note: To ensure that software reset has fully completed and that the Controller responds correctly to subsequent accesses, wait at least 3 milliseconds after setting *CTRL.RST* before attempting to check if the bit was cleared or before attempting to access any other register.

Device Reset (*CTRL.DEV_RST*) can be used to globally reset the entire component, if the *DEV_RST_EN* bit in *Initialization Control 4* EEPROM word is set. This bit is provided as a last-ditch software mechanism to recover from an indeterminate or suspected hardware hung state that could not be resolved by setting the *CTRL.RST* bit. When setting *CTRL.DEV_RST*, most registers (receive, transmit, interrupt, statistics, etc.) and state machines on ports are set to power-on reset values, approximating the state following a power-on or PCI reset. However, PCIe* configuration registers are not reset, leaving the device mapped into system memory space and accessible by a driver.

When *CTRL.DEV_RST* is asserted by software on a LAN port, all LAN ports (including LAN ports that didn't initiate the reset) are placed in a reset state. To notify software device drivers on all ports that *CTRL.DEV_RST* has been asserted, an interrupt is generated and the *ICR.DRSTA* bit is set on all ports.

Following Device Reset assertion or reception of Device Reset interrupt (*ICR.DRSTA*) software should initiate the following steps to re-initialize the port:

1. Wait for the *GCR.DEV_RST in progress* bit to be cleared.
2. Issue a port reset by setting the *CTL.RST* bit, before re-initializing the port. The additional port software reset is required to verify that any Transmit descriptor tail (TDT) updates programmed between device reset and device reset interrupt reception are cleared.
3. Re-initialize the port.

When asserting the *CTRL.RST* software reset bit, only some EEPROM bits related to the specific function are re-read. When setting the *CTRL.DEV_RST* software reset bit, EEPROM bits related to all ports are re-read (see Section 23.3.2.1). Bits re-read from EEPROM are reset to default values.

Fields controlled by the LED, SDP and Init3 words of the EEPROM are not reset and not re-read after a software reset. For the list of words read from EEPROM at full software reset, see Section 23.3.2.1.

23.3.2 BME (Bus Master Enable)

Disabling Bus Master activity of a function by clearing the Configuration *Command register.BME* bit to 0, resets all DMA activities and MSI/MSIx operations related to the port. The Master disable is per function and resets only the DMA activities related to this function without effecting activity of other functions or LAN ports. Configuration accesses and target accesses to the function are still enabled and BMC can still transmit and receive packets on the port.

A Master Disable to a function resets all the queues and DMA related interrupts attached to this function. It also disables the transmit and receive flows for the queues allocated to this function. All pending read requests are dropped and PCIe* read completions to this function might be completed as unexpected completions and silently discarded (following update of flow control credits) without logging or signaling it as an error.

Note: Prior to issuing a master disable the Driver needs to implement the master disable algorithm as defined in Section 25.2.3.2. After Master Enable is set back to 1 driver should re-initialize the transmit and receive queues.

The GbE interface implementation recommends using GIO Master Disable algorithm before disabling BME. Flow when GIO Master Disable is used is shown in "Transaction Flow for BME When GIO Master Disable Algorithm is Not Used". Flow when GIO Master Disable is not used is shown in "Transaction Flow for BME When GIO Master Disable Algorithm is Used".

23.3.2.1 Transaction Flow for BME When GIO Master Disable Algorithm is Used

1. Software sets GIO_MASTER_DISABLE.
2. Software waits for some time and checks if GIO_MASTER_ENABLE_Status is Clear.
   a. Status does not get clear. Go to step 3
   b. Status is clear (No problem exists. Perform BME_Disable). If error was received by Gasket in any of the completed transactions, Interrupt will be asserted by Gasket to EP.
3. Software disables BME to the function.
   a. In response to BME deassertion, EP sends dummy completions for all pending transactions for the affected port.
4. Once GbE receives these dummy completions, it will send completions with bogus data to DMA block for BME-Disabled function.
5. Once all completions for the function have been returned to DMA, GbE Gasket generates GbE_FLR_ACK for the function to EP. This signal GbE_FLR_ACK will play the ACK role for both FLR assertion and BME De-assertion.
6. In future, whenever BME is enabled again for this function, EP will send the pending Interrupt which it might have collected when BME was asserted.

23.3.2.2 Transaction Flow for BME When GIO Master Disable Algorithm is Not Used

1. Software can disable the BME to the function without following GIO_MASTER_DISABLE flow. This is because BME is controlled by OS which may not know about device specific GIO_MASTER_DISABLE register. Same logic applies to FLR where GIO_Master_Disable may not be used.
   a. In response to BME deassertion, EP sends dummy completions for all pending transactions for the affected port.

2. Once GbE receives these dummy completions, it will send completions with bogus data to DMA block for BME-Disabled function.
3. Once all completions for the function have been returned to DMA, GbE Gasket generates GbE_FLR_ACK for the function to EP. This signal GbE_FLR_ACK will play the ACK role for both FLR assertion and BME De-assertion.
4. In future, whenever BME is enabled again for this function, EP will send the pending Interrupt which it might have collected when BME was asserted.

23.3.3 Force TCO

This reset is generated when manageability logic is enabled and BMC detects that the Controller does not receive or transmit data correctly. Force TCO reset is enabled if the *Reset on Force TCO* bit in the *Management Control* EEPROM word is set to 1. Force TCO reset is generated in pass through mode when BMC issues a Force TCO command with bit 1 set and the above conditions exist.

23.3.4 Firmware Reset

This reset is activated by writing a 1 to the FWR bit in the HOST Interface Control Register (HICR) in CSR address 0x8F00.

23.3.5 EEPROM Reset

Writing a 1 to the EEPROM Reset bit of the Extended Device Control Register (*CTRL_EXT.EE_RST*) causes the Controller to re-read the per-function configuration from the EEPROM, setting the appropriate bits in the registers loaded by the EEPROM.

23.3.6 External Phy Reset

To provide the possibility of using Management Unit to program external PHY on power-on and on PHY reset events, a timer is provided that delays the PHY reset completion signal received by Management Unit. This is to allow enough time between power-on, PHY reset and PHY programming through MDIO/I2C interface.

The value of the delay is being programmed in PHY_rst_cmpl_dly CSR through EEPROM on power-on and optionally adjusted by the driver afterwards. The timer counter is halted until the “programming_done” bit in PHY_rst_cmpl_dly is set. This is to prevent signaling MNG until the desired value is set in the CSR. It is required from the Platform Design that external PHY hard reset be derived from the same Power Good signal that is the source for GBE_AUX_POWER_GOOD. This is to provide a common starting point for both external PHY reset and aforementioned timer.

For SW initiated PHY reset the driver sets the PHY_RST bit in Device Control CSR, resets the external PHY through MDIO interface and then clears the PHY_RST bit in Device Control CSR.

23.3.7 CSR Access Flow Following Power-on and Reset Events

Following PCIE_EP_RST#, In-band, FLR and ACPI(D3->D0) resets, HOST will not be able to access GbE Config Space CSRs until the CSR data is valid. In the response to HOST PCIe* CFG requests, PCIe* EP will send config retry response until the above signal is deasserted - per function. MMIO CSRs in the GbE can be only accessesed when the ACPI FSM moves to D0a state - as the effect of setting the appropriate CFG space bits in EP CFG Command Register.

Any agent that wants to access GbE CSR following power-up or PCIE_EP_RST#, In-band, FLR and ACPI resets needs to poll the GBECFGMMIOV register, see Section 20.4.2.5, "GBECFGMMIOV—GBE Configuration and MMIO Valid Register", that reflects the validity of the CSR data. Once the CSR data is validated for a given port the agent is free to access GbE CSRs.

After issuing a software device reset and software port reset any agent needs to wait for at least 3 milliseconds before it can access the GbE MMIO CSRs.

23.3.8 Reset Effects

The resets affect the following registers and logic.

Table 23-2. GbE Controller Reset Effects - Common Resets

Reset Activation	GBE_AUX_ PWR_OK	PCIE_EP_ RST#	SW CTRL. DEV_RST	In-Band PCIe Reset	FW Reset	Notes
LTSSM (PCIe* back to detect/polling)	X	X		X		
PCIe* Link data path	X	X		X		
Read EEPROM (Per Function)			X			17.
Read EEPROM (Complete Load)	X	X		X		
PCI Configuration Registers- non sticky	X	X		X		2.
PCI Configuration Registers - sticky	X	X		X		3.
PCIe* local registers	X	X		X		4.
Data path	X	X	X	X		
On-die memories	X	X	X	X		14.
MAC, PCS, Auto Negotiation, and other port related logic	X	X	X	X		
DMA	X	X	X	X		
Functions queue enable	X	X	X	X		
Function interrupt & statistics registers	X	X	X	X		
Wake Up (PM) Context	X	1.				6.
Wake Up Control Register	X					7.
Wake Up Status Registers	X					8.
Manageability Control Registers	X					9.
MMS Unit	X				X	
Wake-Up Management Registers	X	X	X	X		2.,10.
Memory Configuration Registers	X	X	X	X		2.
EEPROM request	X		X			15.
PHY/SERDES PHY	X	X		X		2.
Strapping Pins	X	X		X		

Table 23-3. GbE Controller Reset Effects - Per Function Resets

Reset Activation	D3hot -> D0	FLR	Port SW Reset (CTRL.RST)	Force TCO	EE Reset		Notes
Read EEPROM (Per Function)	X	X	X	X	X		
PCI Configuration Registers RO							2.
PCI Configuration Registers MSI-X	X	X					5.
PCI Configuration Registers RW							
PCIe* local registers							4.
Data path	X	X	X	X			
On-die memories	X	X	X	X			14.
MAC, PCS, Auto Negotiation, and other port related logic	X	X	X	X			
DMA	X	X					16.
Wake Up (PM) Context							6.
Wake Up Control Register							7.
Wake Up Status Registers							8.
Manageability Control Registers							9.
Function queue enable	X	X	X	X			
Function interrupt & statistics registers	X	X	X				
Wake-Up Management Registers	X	X	X	X			2.,10.
Memory Configuration Registers	X	X	X	X			2.
EEPROM request	X	X					15.
Strapping Pins							

Notes:

1. If GBE_AUX_PWR_AVAIL = 0b the Wakeup Context is reset (*PME_Status* and *PME_En* bits should be 0b at reset if the Controller does not support PME from D3cold).
2. The following register fields do not follow the general rules above:
   a. "CTRL.SDP0_IODIR, CTRL.SDP1_IODIR, CTRL_EXT.SDP2_IODIR, CTRL_EXT.SDP3_IODIR, CONNSW.ENRGSRC field, CTRL_EXT.SFP_Enable, CTRL_EXT.LINK_MODE, CTRL_EXT.EXT_VLAN and LED configuration registers are reset on GBE_AUX_PWR_OK only. Any EEPROM read resets these fields to the values in the EEPROM.
   b. The Aux Power Detected bit in the PCIe* Device Status register is reset on GBE_AUX_PWR_OK and PCIE_EP_RST# (PCIe* reset) assertion only.

c. FLA - reset on GBE_AUX_PWR_OK only.

d. The bits mentioned in the next note.

3. The following registers are part of this group:
   a. VPD registers
   b. Max payload size field in PCIe* Capability Control register (offset 0xA8).
   c. Active State Link PM Control field, Common Clock Configuration field and Extended Synch field in PCIe* Capability Link Control register (Offset 0xB0).
   d. Read Completion Boundary in the PCIe* Link Control register (Offset 0xB0).
4. The following registers are part of this group:
   a. SWSM
   b. *GCR* (only part of the bits - see register description for details)
   c. FUNCTAG
   d. GSCL_1/2/3/4
   e. GSCN_0/1/2/3
   f. *SW_FW_SYNC* - only part of the bits - see register description for details.
5. The following registers are part of this group:
   a. *MSIX control* register, *MSIX PBA* and *MSIX per vector mask*.
6. The Wake Up Context is defined in the PCI Bus Power Management Interface Specification (Sticky bits). It includes:
   — *PME_En* bit of the Power Management Control/Status Register (*PMCSR*).
   — *PME_Status* bit of the Power Management Control/Status Register (*PMCSR*).
   — *Aux_En* in the PCIe* registers
   — The device Requester ID (since it is required for the PM_PME TLP).

   The shadow copies of these bits in the Wakeup Control Register are treated identically.
7. Refers to bits in the Wake Up Control Register that are not part of the Wake-Up Context (the *PME_En* and *PME_Status* bits).
8. Wake Up Status registers:
   a. Wake Up Status Register
   b. Wake Up Packet Length.
   c. Wake Up Packet Memory.
9. Manageability control registers:
   a. *MANC* 0x5820
   b. *MFUTP0-7* 0x5030 - 0x504C
   c. *MNGONLY* 0x5864
   d. *MAVTV0-7* 0x5010 - 0x502C
   e. *MDEF0-7* 0x5890 - 0x58AC
   f. *MDEF_EXT* 0x5930 - 0x594C
   g. *METF0-3* 0x5060 - 0x506C
   h. *MIPAF0-15* 0x58B0 - 0x58EC
   i. *MMAH/MMAL0*-1 0x5910 - 0x591C
   j. FWSM

10. Wake-up Management registers:
    a. Wake Up Filter Control
    b. IP Address Valid
    c. IPv4 Address Table
    d. IPv6 Address Table
    e. Flexible Filter Length Table
    f. Flexible Filter Mask Table
11. Other configuration registers:
    a. General Registers
    b. Interrupt Registers
    c. Receive Registers
    d. Transmit Registers
    e. Statistics Registers
    f. Diagnostic Registers

Of these registers, *MTA[n], VFTA[n], WUPM[n], FTFT[n], FHFT[n], FHFT_EXT[n], TDBAH/TDBAL*, and *RDBAH/RDVAL* registers have no default value. If the functions associated with the registers are enabled they must be programmed by software. Once programmed, their value is preserved through all resets as long as power is applied to the Controller.

Note: In situations where the device is reset using the software reset *CTRL.RST* or *CTRL.DEV_RST* the transmit data lines are forced to all zeros. This causes a substantial number of symbol errors to be detected by the link partner. In TBI mode, if the duration is long enough, the link partner might restart the Auto-Negotiation process by sending “break-link” (/C/ codes with the configuration register value set to all zeros).

12. These registers include:
    a. MSI/MSI-X enable bits
    b. BME
    c. Error indications
13. These registers include:
    a. RXDCTL.Enable
14. The contents of the following memories are cleared to support the requirements of PCIe* FLR:
    a. The Tx packet buffers
    b. The Rx packet buffers
15. Includes *EEC.REQ*, *EEC.GNT*, *FLA.REQ* and *FLA.GNT* fields.
16. The following DMA Registers are cleared only by GBE_AUX_PWR_OK, PCIe* Reset or *CTRL.DEV_RST*: *DMCTLX, DTPARS, DRPARS* and *DDPARS.*
17. *CTRL.DEV_RST* assertion causes read of function related sections for all ports

23.4 Software Initialization and Diagnostics

This chapter discusses general software notes for the Controller, especially initialization steps. This includes general hardware, power-up state, basic device configuration, initialization of transmit and receive operation, link configuration, software reset capability, statistics, and diagnostic hints.

23.4.1 Power Up State

When the Controller powers up, it reads the EEPROM. The EEPROM contains sufficient information to bring the link up and configure the Controller for manageability and/or APM wakeup. However, software initialization is required for normal operation.

The power-up sequence, as well as transitions between power states, are described in section 23.1.1. The detailed timing is given in Section 25.3. The next section gives more details on configuration requirements.

23.4.2 Initialization Sequence

The following sequence of commands is typically issued to the device by the software device driver in order to initialize the Controller to normal operation. The major initialization steps are:

- Disable Interrupts - see Interrupts during initialization.
- Issue Global Reset and perform General Configuration - see Global Reset and General Configuration.
- Setup the PHY and the link - see Link Setup Mechanisms and Control/Status Bit Summary.
- Initialize all statistical counters - see Initialization of Statistics.
- Initialize Receive - see Receive Initialization.
- Initialize Transmit - see Transmit Initialization.
- Enable Interrupts - see Interrupts during initialization.

23.4.3 Interrupts During Initialization

- Most drivers disable interrupts during initialization to prevent re-entering the interrupt routine. Interrupts are disabled by writing to the *EIMC* (Extended Interrupt Mask Clear) register. The interrupts need to be disabled also after issuing a global reset, so a typical driver initialization flow is:
- Disable interrupts
- Issue a Global Reset
- Disable interrupts (again)
- …

After the initialization is done, a typical driver enables the desired interrupts by writing to the *EIMS* (Extended Interrupt Mask Set) register.

23.4.4 Global Reset and General Configuration

Device initialization typically starts with a global reset that places the device into a known state and enables the device driver to continue the initialization sequence.

Several values in the Device Control Register (*CTRL*) need to be set, upon power up, or after a device reset for normal operation.

- *FD* bit should be set per interface negotiation (if done in software), or is set by the hardware if the interface is Auto-Negotiating. This is reflected in the *Device Status* Register in the Auto-Negotiation case.
- Speed is determined via Auto-Negotiation by the PHY, Auto-Negotiation by the PCS layer in SGMII/SerDes mode, or forced by software if the link is forced. Status information for speed is also readable in the *STATUS* register.
- *ILOS* bit should normally be set to 0.

23.4.5 Flow Control Setup

If flow control is enabled, program the *FCRTL0*, *FCRTH0*, *FCTTV* and *FCRTV* registers. In order to avoid packet losses, *FCRTH* should be set to a value equal to at least two max size packet below the receive buffer size. For example, Assuming a packet buffer size of 32 KB and expected max size packet of 9.5K, the *FCRTH0* value should be set to 32 - 2 * 9.5 = 17KB for example *FCRTH0.RTH* should be set to 0x440.

23.4.6 Link Setup Mechanisms and Control/Status Bit Summary

The *CTRL_EXT.LINK_MODE* value should be set to the desired mode prior to the setting of the other fields in the link setup procedures.

23.4.6.1 MAC/SERDES Link Setup (CTRL_EXT.LINK_MODE = 011b)

Link setup procedures using an external SERDES interface mode:

23.4.6.1.1 Hardware Auto-Negotiation Enabled (PCS_LCTL. AN ENABLE = 1b)

CTRL.FD	Ignored; duplex is set by priority resolution of *PCS_ANDV* and *PCS_LPAB*.
CTRL.SLU	Must be set to 1 by software to enable communications to the SerDes.
CTRL.RFCE	Set by Hardware according to auto negotiation resolution[1].
CTRL.TFCE	Set by Hardware according to auto negotiation resolution[1].
CTRL.SPEED	Ignored; speed always 1000Mb/s when using SerDes mode communications.
STATUS.FD	Reflects hardware-negotiated priority resolution.
STATUS.LU	Reflects *PCS_LSTS.AN COMPLETE* (Auto-Negotiation complete).
STATUS.SPEED	Reflects 1000Mb/s speed, reporting fixed value of (10)b.
PCS_LCTL.FSD	Must be zero.
PCS_LCTL.Force Flow Control	Must be zero[1].

1. If PCS_LCTL.Force Flow Control is set, the auto negotiation result is not reflected in the CTRL.RFCE and CTRL.TFCE registers. In This case, the software must set these fields after reading flow control resolution from PCS registers.

PCS_LCTL.FSV	Must be set to 10b. Only 1000 Mb/s is supported in SerDes mode.
PCS_LCTL.FDV	Ignored; duplex is set by priority resolution of *PCS_AND*V and *PCS_LPAB*.

23.4.6.1.2 Auto-Negotiation Skipped (PCS_LCTL. AN ENABLE = 0b)

CTRL.FD	Must be set to 1b. - only full duplex is supported in SerDes mode.
CTRL.SLU	Must be set to 1 by software to enable communications to the SerDes.
CTRL.RFCE	Set by software for the desired mode of operation.
CTRL.TFCE	Set by software for the desired mode of operation.
CTRL.SPEED	Must be set to 10b. Only 1000 Mb/s is supported in SerDes mode.
STATUS.FD	Reflects the value written by software to *CTRL.FD*.
STATUS.LU	Reflects whether the PCS detected comma symbols, qualified with *CTRL.SLU* (set to 1).
STATUS.SPEED	Reflects 1000Mb/s speed, reporting fixed value of (10)b.
PCS_LCTL.FSD	Must be set to 1 by software to enable communications to the SerDes.
PCS_LCTL.Force Flow Control	Must be set to 1.
PCS_LCTL.FSV	Must be set to 10b. Only 1000 Mb/s is supported in SerDes mode.
PCS_LCTL.FDV	Must be set to 1b - only full duplex is supported in SerDes mode.

23.4.6.2 MAC/SGMII Link Setup (CTRL_EXT.LINK_MODE = 010b)

Link setup procedures using an external SGMII interface mode:

23.4.6.2.1 Hardware Auto-Negotiation Enabled (PCS_LCTL. AN ENABLE = 1b, CTRL.FRCDPLX = 0b, CTRL.FRCSPD = 0b)

CTRL.FD	Ignored; duplex is set by priority resolution of PCS_ANDV and PCS_LPAB.
CTRL.SLU	Must be set to 1 by software to enable communications to the SerDes.
CTRL.RFCE	Must be set by software after reading flow control resolution from PCS registers.
CTRL.TFCE	Must be set by software after reading flow control resolution from PCS registers.
CTRL.SPEED	Ignored; speed setting is established from SGMII's internal indication to the MAC after SGMII PHY has auto-negotiated a successful link-up.
STATUS.FD	Reflects hardware-negotiated priority resolution.
STATUS.LU	Reflects *PCS_LSTS.Link OK*
STATUS.SPEED	Reflects actual speed setting negotiated by the SGMII and indicated to the MAC.

PCS_LCTL.Force Flow Control	Ignored.
PCS_LCTL.FSD	Should be set to zero.
PCS_LCTL.FSV	Ignored; speed is set by priority resolution of *PCS_ANDV* and *PCS_LPAB*.
PCS_LCTL.FDV	Ignored; duplex is set by priority resolution of *PCS_ANDV* and *PCS_LPAB*.

23.4.6.3 MAC/1000BASE-KX Link Setup (CTRL_EXT.LINK_MODE = 001b)

23.4.6.3.1 Auto-Negotiation Skipped (PCS_LCTL. AN ENABLE = 0b; CTRL.FRCSPD = 1b; CTRL.FRCDPLX = 1)

Link setup procedures using an external 1000BASE-KX Server Backplane interface mode:

CTRL.FD	Must be set to 1. 1000BASE-KX always in full duplex mode.
CTRL.SLU	Must be set to 1 by software to enable communications to the SerDes.
CTRL.RFCE	Must be set by software for the desired mode of operation.
CTRL.TFCE	Must be set by software for the desired mode of operation.
CTRL.SPEED	Must be set to 10b. Only 1000 Mb/s is supported in 1000BASE-KX mode.
STATUS.FD	Reflects the value written by software to *CTRL.FD*.
STATUS.LU	Reflects whether the PCS detected comma symbols, qualified with *CTRL.SLU* (set to 1).
STATUS.SPEED	Reflects 1000Mb/s speed, reporting fixed value of (10b).
PCS_LCTL.FSD	Must be set to 1 by software to enable communications to the 1000BASE-KX SerDes.
PCS_LCTL.Force Flow Control	Must be set to 1.
PCS_LCTL.FSV	Must be set to 10b. Only 1000 Mb/s is supported in 1000BASE-KX mode.
PCS_LCTL.FDV	Must be set to 1b - only full duplex is supported in 1000BASE-KX mode.

23.4.7 Initialization of Statistics

Statistics registers are hardware-initialized to values as detailed in each particular register's description. The initialization of these registers begins upon transition to D0 active power state (when internal registers become accessible, as enabled by setting the Memory Access Enable of the PCIe* Command register), and is guaranteed to be completed within 1 μsec of this transition. Access to statistics registers prior to this interval might return indeterminate values.

All of the statistical counters are cleared on read and a typical device driver reads them (thus making them zero) as a part of the initialization sequence.

23.4.8 Receive Initialization

Program the Receive address register(s) per the station address. This can come from the EEPROM or by any other means (for example, on some machines, this comes from the system PROM not the EEPROM on the adapter card).

Set up the *MTA* (Multicast Table Array) by software. This means zeroing all entries initially and adding in entries as requested.

Program *RCTL* with appropriate values. If initializing it at this stage, it is best to leave the receive logic disabled (*EN* = 0b) until after the receive descriptor rings have been initialized. If VLANs are not used, software should clear *VFE*. Then there is no need to initialize the *VFTA*. Select the receive descriptor type.

The following should be done once per receive queue needed:

- Allocate a region of memory for the receive descriptor list.
- Receive buffers of appropriate size should be allocated and pointers to these buffers should be stored in the descriptor ring.
- Program the descriptor base address with the address of the region.
- Set the length register to the size of the descriptor ring.
- Program *SRRCTL* of the queue according to the size of the buffers, the required header handling and the drop policy.
- If header split or header replication is required for this queue, program the *PSRTYPE* register according to the required headers.
- Enable the queue by setting *RXDCTL.ENABLE*. In the case of queue zero, the enable bit is set by default - so the ring parameters should be set before *RCTL.RXEN* is set.
- Poll the *RXDCTL* register until the *ENABLE* bit is set. The tail should not be bumped before this bit was read as one.
- Program the direction of packets to this queue according to the mode selected in the *MRQC* register. Packets directed to a disabled queue are dropped.

Note: The tail register of the queue (*RDT[n]*) should not be bumped until the queue is enabled.

23.4.8.1 Initialize the Receive Control Register

To properly receive packets the receiver should be enabled by setting RCTL.RXEN. This should be done only after all other setup is accomplished. If software uses the Receive Descriptor Minimum Threshold Interrupt, that value should be set.

23.4.8.2 Dynamic Enabling and Disabling of Receive Queues

Receive queues can be dynamically enabled or disabled given the following procedure is followed:

Enabling:

- Follow the per queue initialization described in the previous section.

Disabling:

- Disable the direction of packets to this queue.
- Disable the queue by clearing *RXDCTL.ENABLE*. The Controller stops fetching and writing back descriptors from this queue immediately. The Controller eventually completes the storage of one buffer allocated to this queue. Any further packet

directed to this queue is dropped. If the currently processed packet is spread over more than one buffer, all subsequent buffers are not written.

- The Controller clears *RXDCTL.ENABLE* only after all pending memory accesses to the descriptor ring or to the buffers are done. The driver should poll this bit before releasing the memory allocated to this queue.

The RX path might be disabled only after all Rx queues are disabled.

23.4.9 Transmit Initialization

- Program the *TCTL* register according to the MAC behavior needed.

If work in half duplex mode is expected, program the *TCTL_EXT.COLD* field. a value reflecting the Controller and the PHY SGMII delays should be used. A suggested value for a typical PHY is 0x46 for 10 Mbps and 0x4C for 100 Mbps.

The following should be done once per transmit queue:

- Allocate a region of memory for the transmit descriptor list.
- Program the descriptor base address with the address of the region.
- Set the length register to the size of the descriptor ring.
- Program the *TXDCTL* register with the desired TX descriptor write back policy. Suggested values are:
  - *WTHRESH* = 1b
  - All other fields 0b.
- If needed, set the *TDWBAL/TWDBAH* to enable head write back
- Enable the queue using *TXDCTL.ENABLE* (queue zero is enabled by default).
- Poll the *TXDCT*L register until the *ENABLE* bit is set.

Note: The tail register of the queue (*TDT[n]*) should not be bumped until the queue is enabled.

Enable transmit path by setting *TCTL.EN*. This should be done only after all other settings are done.

23.4.9.1 Dynamic Queue Enabling and Disabling

Transmit queues can be dynamically enabled or disabled given the following procedure is followed:

Enabling:

- Follow the per queue initialization described in the previous section.

Disabling:

- Stop storing packets for transmission in this queue.
- Wait until the head of the queue (*TDH)* is equal to the tail (*TDT*), for example the queue is empty.
- Disable the queue by clearing *TXDCTL.ENABLE*.

The Tx path might be disabled only after all Tx queues are disabled.

23.4.10 Virtualization Initialization Flow

23.4.10.1 VMDq Mode

23.4.10.1.1 Global Filtering and Offload Capabilities

- Select the VMDQ pooling method - MAC/VLAN filtering for pool selection. *MRQC.Multiple Receive Queues Enable* = 011b.
- Set the *RPLOLR* and *RPLPSRTYPE* registers to define the behavior of replicated packets.
- Configure *VT_CTL.DEF_PL* to define the default pool. If packets with no pools should be dropped, set *VT_CTL.Dis_def_Pool* field.
- If needed, enable padding of small packets via the *RCTL.PSP*

23.4.10.1.2 Per Pool Settings

As soon as a pool of queues is associated to a VM the software should set the following parameters:

1. Address filtering:
   a. The unicast MAC address of the VM by enabling the pool in the *RAH/RAL* registers.
   b. If all the MAC addresses are used, the unicast hash table (*UTA*) can be used. Pools servicing VMs whose address is in the hash table should be declared as so by setting the *VMOLR.ROPE*. Packets received according to this method didn't pass perfect filtering and are indicated as such.
   c. Enable the pool in all the *RAH*/*RA*L registers representing the multicast MAC addresses this VM belongs to.
   d. If all the MAC addresses are used, the multicast hash table (*MTA*) can be used. Pools servicing VMs using multicast addresses in the hash table should be declared as so by setting the *VMOLR.ROMPE*. Packets received according to this method didn't pass perfect filtering and are indicated as such.
   e. Define whether this VM should get all multicast/broadcast packets in the same VLAN via the *VMOLR.MPE* and *VMOLR.BAM fields*
   f. Enable the pool in each *VLVF* register representing a VLAN this VM belongs to.
   g. Define whether the pool belongs to the default VLAN and should accept untagged packets via the *VMOLR.AUPE* field
2. Offloads
   a. Define whether VLAN header should be stripped from the packet (defined by *VMOLR.strvlan*).
   b. Set which header split is required via the *PSRTYPE* register.
   c. Set whether larger than standard packet are allowed by the VM and what is the largest packet allowed (jumbo packets support) via *VMOLR.*RLPML & *VMOLR.RLE*.
3. Queues
   a. Enable Rx & Tx queues as described in Section 23.4.8 & Section 23.4.9
   b. For each Rx queue a drop/no drop flag can be set in *SRRCTL.DROP_EN*, controlling the behavior in cases no receive buffers are available in the queue to receive packets. The usual behavior is to allow drops in order to avoid head of line blocking, unless a no-drop behavior is needed for some type of traffic (for example, storage).

23.4.10.1.3 Security Features: Storm Control

The driver may set limits to the broadcast or multicast traffic it can receive.

1. It should set the how many 64 byte chunks of Broadcast and Multicast traffic are acceptable per interval via the *BSCTRH* and *MSCTRH* respectively.
2. It should then set the interval to be used via the *SCCRL.Interval* field and which action to take when the broadcast or multicast traffic crosses the programmed threshold via the *SCCRL.BDIPW, SCCRL.BDICW, SCCRL.MDIPW,* and *SCCRL.MDICW* fields.

- The driver may be notified of storm control events through the *ICR.SCE* interrupt cause.

23.5 Access to Shared Resources

Part of the resources for the GbE interface in the PCH are shared between several software entities - namely the drivers of the four ports and the internal firmware. In order to avoid contentions, a driver that needs to access one of these resources should use the flow described in Section 23.5.1 in order to acquire ownership of this resource and use the flow described in Section 23.5.2 in order to relinquish ownership of this resource.

The shared resources are:

1. EEPROM.
2. All SerDes ports.
3. CSRs accessed by the internal firmware after the initialization process. Currently there are no such CSRs.
4. Software to Software Mailbox

Note: Any other software tool that accesses the Controller register set directly should also follow the flow described below.

23.5.1 Acquiring Ownership Over a Shared Resource

The following flow should be used to acquire a shared resource:

a. If *SWSM.SMBI* is read as zero, the semaphore was taken.

b. Otherwise, repeat steps.

SW_FW_SYNC

a. Set the bit.

b. Read If was successfully set - the semaphore was acquired - otherwise, go back to step a.

SW_FW_SYNC

Software reads the Software-Firmware Synchronization Register (SW_FW_SYNC) and checks both bits in the pair of bits that control the resource it wishes to own.

a. If both bits are cleared (both firmware and other software does not own the resource), software sets the software bit in the pair of bits that control the resource it wishes to own.

b. If one of the bits is set (firmware or other software owns the resource), software tries again later.

At this stage, the shared resources is owned by the driver and it may access it. The *SWSM* and *SW_FW_SYNC* registers can now be used to take ownership of another shared resources.

Note: Bit should not exceed 100 mS. If software takes ownership for a longer duration, Firmware may implement a timeout mechanism and take ownership.

Software ownership of bits in *SW_FW_SYNC* register should not exceed 1 second. If Software takes ownership for a longer duration, Firmware may implement a timeout mechanism and take ownership of the relevant *SW_FW_SYNC* bits.

23.5.2 Releasing Ownership Over a Shared Resource

The following flow should be used to release a shared resource:

a. If *SWSM.SMBI* is read as zero, the semaphore was taken.

b. Otherwise, go back to step a.

SW_FW_SYNC

a. Set the bit.

b. Read If was successfully set - the semaphore was acquired - otherwise, go back to step a.

SW_FW_SYNC

Clear the bit in *SW_FW_SYNC* that controls the software ownership of the resource to indicate this resource is free.

At this stage, the shared resource are released by the driver and it may not access it. The *SWSM* and *SW_FW_SYNC* registers can now be used to take ownership of another shared resource.

§ §

24.0 Non-Volatile Memory Map - EEPROM

24.1 EEPROM General Map

This section pertains to the DH89xxCC devices only.

The EEPROM for the DH89xxCL devices must use the supplied configuration image. The EEPROM for the DH89xxCL cannot be configured via the IA. If an EEPROM programming functionality is needed, it must be implemented at the platform level.

The GbE controller EEPROM is partitioned into four main blocks followed by Firmware and PXE structures. The first 128 word section is allocated to words common to all LAN ports, Firmware, Software, PXE and LAN port 0. The following 64 word sections are allocated to LAN port 1, LAN port 2 and LAN port 3 as shown in Table 24-1.

On a software reset (*CTRL.RST* or *CTRL.DEV_RST*) only the words defined in the auto load sequence in Section 22.3.1.3 are re-loaded by HW. On power-up, Hardware reset or *CTRL_EXT.EE_RS*T assertion all words are loaded from EEPROM.

Table 24-1. EEPROM Top Level Partitioning

EEPROM Word Address	Partition
0x0 to 0x7F	Common Words (PCIe*, PXE, SW and FW) and LAN port 0 words
0x80 to 0xBF	LAN Port 1 words
0xC0 to 0xFF	LAN Port 2 words
0x100 to 0x13F	LAN Port 3 words
0x140 to …	PXE and FW structures

Table 24-2 lists the GbE controller EEPROM word map for Common words and Lan Port 0.

Note: Reserved EEPROM entries can contain any value, except where noted.

Table 24-2. Common and Lan Port 0 EEPROM Map (Sheet 1 of 3)

EEPROM Word Address	Used By/In	High Byte	Low Byte	Which LAN
0	HW	Ethernet Address Byte 2	Ethernet Address Byte 1	LAN 0
01	HW	Ethernet Address Byte 4	Ethernet Address Byte 3	LAN 0
02	HW	Ethernet Address Byte 6	Ethernet Address Byte 5	LAN 0
03	SW	Compatibility High	Compatibility Low	All
04		Reserved	Reserved	
05		Compatibility High	Compatibility Low	All
06 - 07		Reserved	Reserved	
08	SW	Reserved (must default to FAFAh)		All

Table 24-2. Common and Lan Port 0 EEPROM Map (Sheet 2 of 3)

EEPROM Word Address	Used By/In	High Byte	Low Byte	Which LAN
09	SW	Pointer to PBA Structure		All
0A	HW	Init Control 1		All
0B	HW	Subsystem ID		All
0C	HW	Subsystem Vendor ID		All
0D	HW	Device ID		LAN 0
0E	HW	Vendor ID		All
0F	HW	Init Control 2		All
10		Reserved (Value should be FFFFh)		
11	HW (MNG HW)	Pass Through LAN Configuration Pointer		LAN 0 (MNG HW)
12	HW	EEPROM Sizing		All
13	HW	Init Control 4		LAN 0
14:15		Reserved		
16	HW	GbE Controller: (PHY_RST_Control) • 15:4 --> Reserved • 3 --> Phy_Rst_Cmpl_Prog_Done • 2:0 --> Phy_Rst_Cmpl_Dly		LAN 0
17	HW	CSR Auto Configuration Pointer		LAN 0
18:1D		Reserved		
1E	HW	Device REV ID		All
1F	HW	LEDCTL 0 Default		LAN 0
20	HW	Software Defined Pins Control		LAN 0
21	HW	CGB Functions Control		All
22		Reserved		
23	HW	Management Hardware Config Control		LAN 0 (MNG HW)
24	HW	Init Control 3		LAN 0
25-2D		Reserved		All
2E	HW	Watchdog Configuration		All
2F	OEM	VPD Pointer		All
30	PXE	PXE main setup options		LAN0
31	PXE	PXE Configuration Customization options		LAN0
32		Reserved (Value should be FFFFh)		
33		Reserved (Value should be FFFFh)		
34	PXE	PXE main setup options		LAN1
35	PXE	PXE Configuration Customization options		LAN1
36		Reserved (Value should be FFFFh)		
37		Pointer to Alternate MAC address		All
38	PXE	PXE main setup options		LAN2
39	PXE	PXE Configuration Customization options		LAN2
3A	PXE	PXE main setup options		LAN3
3B	PXE	PXE Configuration Customization options		LAN3

Table 24-2. Common and Lan Port 0 EEPROM Map (Sheet 3 of 3)

EEPROM Word Address	Used By/In	High Byte	Low Byte	Which LAN
3C		Reserved (Value should be FFFFh)		
3D		Reserved (Value should be FFFFh)		
3E		Reserved (Value should be FFFFh)		
3F	SW	Software Checksum, Words 00h - 3Eh		
40:41	SW	Reserved		
42	SW	Image Unique ID Low		
43	SW	Image Unique ID High		
44:4F		Reserved		
50	FW	PHY Configuration Pointer (Word 0x50)		All MNG
51	FW	Firmware patch Pointer		MNG Code
52:53		Reserved		MNG
54	FW	MNG Capabilities		MNG HW
55:56		Reserved		MNG
57	FW	Sideband Configuration Pointer		MNG HW
58:5D		Reserved		MNG
5E:7F		Reserved		MNG
80:BF		LAN Port 1 words		
C0:FF		LAN Port2 words		
100:13F		LAN Port3 words		
140	FW/PXE	Firmware and PXE Structures		

Table 24-3 maps the GbE controller EEPROM words that can hold different content for LAN Ports 0, 1, 2 and 3. Addresses listed in the table are an offset from the LAN Base address of the relevant EEPROM LAN section. EEPROM LAN Base addresses of the LAN ports are as follows:

- LAN Port 0 EEPROM section Base Address - 0x0
- LAN Port 1 EEPROM section Base Address – 0x80
- LAN Port 2 EEPROM section Base Address – 0xC0
- LAN Port 3 EEPROM section Base Address – 0x100

Note: Reserved EEPROM entries can contain any value, except where noted.

Table 24-3. LAN Ports 1, 2, and 3 EEPROM Map (Sheet 1 of 2)

EEPROM Word Offset	Used By/In	High Byte	Low Byte
0	HW	Ethernet Address Byte 2	Ethernet Address Byte 1
01	HW	Ethernet Address Byte 4	Ethernet Address Byte 3
02	HW	Ethernet Address Byte 6	Ethernet Address Byte 5
03:0C	HW	Reserved	
0D	HW	Device ID	

Table 24-3. LAN Ports 1, 2, and 3 EEPROM Map (Sheet 2 of 2)

EEPROM Word Offset	Used By/In	High Byte	Low Byte
0E:10	HW	Reserved	
11	HW (MNG HW)	Pass Through LAN Configuration Pointer	
12	HW	Reserved	
13	HW	Init Control 4	
14: 16	HW	Reserved	
17	HW	CSR Auto Configuration Pointer	
18:1B	HW	Reserved	
1C	HW	Reserved	
1D:1E	HW	Reserved	
1F	HW	LEDCTL 0 Default	
20	HW	Software Defined Pins Control	
21	HW	CGB Functions Control	
22:23	HW	Reserved	
24	HW	Init Control 3	
25:26	HW	Reserved	
27	HW	CSR Auto Configuration Power-Up Pointer	
28:29	HW	Reserved	
2A:3E	HW	Reserved	
3F	SW	Software Checksum, Words 00h - 3Eh	

24.2 EEPROM Configuration Notes

Setup and configuration of the PCH GbE MAC has the following dependencies which must be considered when programming the EEPROM.

- The Intel Network Boot drivers and Intel GbE drivers assumes the EEPROM contains the following MAC configuration details:
  - Connection 'Type': The configuration details associated with the MAC connection type (SGMII/SerDes) must be programmed to the EEPROM.
  - Auto-Negotiation: Auto-Negotiation must be programmed correctly for the MAC configuration via the Auto-Negotiation Enable (ANE) bit. ANE is contained within Initialization Control Word 2 which is one of the common words as listed in Figure 24-2, "Common and Lan Port 0 EEPROM Map" on page 1059. The common words are common across all four MACs therefore auto-negotiation is either enabled or disabled across all MACs.
  - PCS parallel detect: PCS parallel detect must be correctly programmed for the MAC configuration via the PCS parallel detect bit. PCS parallel detect is contained within Initialization Control Word 2 which is one of the common words as listed in Figure 24-2, "Common and Lan Port 0 EEPROM Map" on page 1059. The common words are common across all four MACs therefore PCS parallel detect is either enabled or disabled across all MACs.

Since the common words are shared across the four MACs, the EEPROM configuration of the four MACs is not entirely independent. The supported connection types, as configured from the EEPROM, are listed in Figure 24-4, "Supported MAC Connection Types" on page 1063.

Table 24-4. Supported MAC Connection Types

Link Mode	ANE	PCS	Comment
All enabled MACs in SGMII	1	0	
All enabled MACs in SerDes KX	0	1	No ANE
All enabled MACs in SerDes BX	1	0	ANE
All enabled MACs in SerDes BX	0	1	No ANE
All enabled MACs in SGMII and SerDes BX ANE	1	0	ANE. The MACs are mixed between SGMII and SerDes.

24.3 Hardware Accessed Words

This section describes the EEPROM words that are loaded by the GbE controller hardware. Most of these bits are located in configuration registers. The words are only read and used if the signature field in the *EEPROM Sizing* EEPROM word (word 0x12) is valid.

Note: When **Word** is mentioned before an EEPROM address, address is the absolute address in the EEPROM. When **Offset** is mentioned before an EEPROM address, the address is relative to the start of the relevant EEPROM section.

24.3.1 Ethernet Address (LAN Base Address + Offsets 0x00-0x02)

The Ethernet Individual Address (IA) is a 6-byte field that must be unique for each NIC, and thus unique for each copy of the EEPROM image. The first three bytes are vendor specific. For example, the IA is equal to [00 AA 00] or [00 A0 C9] for Intel products. The value from this field is loaded into the Receive Address Register 0 (RAL0/RAH0).

For the purpose of this specification, the IA byte numbering convention is indicated as follows:

	IA Byte / Value					
Vendor	**1**	**2**	**3**	**4**	**5**	**6**
Intel Original	00	AA	00	variable	variable	variable
Intel New	00	A0	C9	variable	variable	variable

The Ethernet address is loaded for LAN0 from Addresses 0x0 to 0x02 and for LAN 1, 2 and 3 from offsets 0x0 to 0x2 at the start of the relevant sections.

24.3.2 Initialization Control Word 1 (Word 0x0A)

The *Initialization Control Word 1* in the Common section contains initialization values that:

- Set defaults for some internal registers
- Enable/disable specific features
- Determine which PCI configuration space values are loaded from the EEPROM

Bit	Name	Default Value in EEPROM	Description
15:14	Reserved	0x0	Reserved
13	Reserved	0b	Reserved.
12	Reserved	0b	Reserved
11	FRCSPD	0b	Default setting for the *Force Speed* bit in the Device Control register (CTRL[11]). See Section 28.1.4.2 . Bit must always be 0b..
10	FD	0b	This bit sets the FRCDPLX (bit 12) of the "Device Control Register - CTRL [0:3] (0x00000; R/W)": 0x0 – All enabled Ports are in SGMII mode. 0x1 – Any enabled Port is in SerDes mode Note: If mixing SerDes and SGMII modes set FD to 1'b1 in the EEPROM word Init_CTRL_1. For all ports in SGMII mode must use EEPROM "CSR Auto Config Ptr (LANx + 0x17)" to set FRCDPLX to 0 in "Device Control Register - CTRL [0:3] (0x00000; R/W)".
9:7	Reserved	11b	Reserved.
6	SDP_IDDQ_EN	0b	When set, SDP IOs keep their value and direction when the controller enters dynamic IDDQ mode either due to PCIe* entering Dr state. Otherwise, SDP IOs moves to HighZ + pull-up mode in dynamic IDDQ mode. Reflected in EEDIAG (See Section 28.3.1.3).
5	Deadlock Timeout Enable	1b	If set, a device granted access to the EEPROM that does not toggle the interface for more than 2 seconds will have the grant revoked. See Section 22.3.2.1.
4	Reserved	0b	Reserved.
3	Power Management	1b	0b = Reserved 1b = Full support for power management (For normal operation, this bit must be set to 1b).
2	Reserved	0b	Reserved
1	Load Subsystem IDs	1b	When this bit is set to 1b the controller loads its PCIe* Subsystem ID and Subsystem Vendor ID from the EEPROM (*Subsystem ID* and *Subsystem Vendor ID* EEPROM words).
0	Load Vendor/Device IDs	1b	When set to 1b the controller loads its PCIe* Device IDs from the EEPROM (*Device ID* EEPROM words) and the PCIe* Vendor ID from the EEPROM.

24.3.3 Subsystem ID (Word 0x0B)

If the *Load Subsystem IDs* in *Initialization Control Word 1* EEPROM word is set, the *Subsystem ID* word in the Common section is read in to initialize the PCIe* Subsystem ID. Default value is 0x0.

24.3.4 Subsystem Vendor ID (Word 0x0C)

If the *Load Subsystem IDs* bit in *Initialization Control Word 1* EEPROM word is set, the Subsystem Vendor ID word in the Common section is read in to initialize the PCIe* Subsystem Vendor ID. The default value is 0x8086.

24.3.5 Device ID (LAN Base Address + Offset 0x0D)

If the *Load Vendor/Device IDs* bit in *Initialization Control Word 1* is set, the *Device ID* EEPROM word is read in from the Common, LAN 1, LAN 2 and LAN 3 sections to initialize the Device ID of LAN0, LAN1, LAN2 and LAN3 functions, respectively. The default value is 0x10C9.

24.3.6 Vendor ID (Word 0x0E)

If the *Load Vendor/Device IDs* bit in *Initialization Control Word 1* EEPROM word is set, this word is read in to initialize the PCIe* Vendor ID. The default value is 0x8086.

Note: If a value of 0xFFFF is placed in the *Vendor ID* EEPROM word, the value in the PCIe* *Vendor ID* register will return to the default 0x8086 value. This functionality is implemented to avoid a system hang situation.

24.3.7 Initialization Control Word 2 (Word 0x0F)

The *Initialization Control Word 2* read by the GbE controller, contains additional initialization values that:

- Set defaults for some internal registers
- Enable/disable specific features

Bit	Name	Default Value in EEPROM	Description
15	APM PME# Enable	0b	Initial value of the *Assert PME On APM Wakeup* bit in the Wake Up Control (WUC.APMPME) register. See Section 28.20.1.1 .
14	PCS parallel detect	1b	**Controls PCS parallel detection:** SGMII = All enabled media interfaces are in SGMII mode set to 0x0. SerDes & SGMII = Both SerDes and SGMII media interfaces are active set to 0x0. SerDes = All enabled media interfaces are in SerDes mode set to 0x0 or 0x1 depending on your system. Mapped to PCS_LCTL.AN TIMEOUT EN bit. PCS Link Control - PCS_LCTL [0:3] (0x4208; RW). See Section 28.17.1.2 .
13:12	Pause Capability	11b	Desired pause capability for advertised configuration base page. Mapped to *PCS_ANADV.ASM*. See Section 28.17.1.4 .
11	ANE	0b	Auto-Negotiation Enable Mapped to *PCS_LCTL.AN_ENABLE*. See Section 28.17.1.2
10:8	Reserved		Reserved
7	MAC clock gating enable	1b	Enables MAC clock gating power saving mode. Mapped to *STATUS[31]*. This bit is relevant only if the *Enable Dynamic MAC Clock Gating* bit is set. See Section 28.1.4.3. Note: Bit impacts clock gating of both regular and Wake-up MAC clocks.
6	Reserved	0b	Reserved and RO as “0”
5	Reserved		Reserved
4	Reserved	0b	Reserved and RO as “0”
3	Enable Dynamic MAC Clock Gating	0b	When set, enables dynamic MAC clock gating mechanism. See Section 28.1.4.4.
2	SerDes Low Power Enable	0b	When set, enables the SerDes to enter a low power state when the function is in Dr state. See Chapter 25.0 and Section 28.1.4.4.
1	Reserved.	0b	Reserved.
0	Reserved	0b	Reserved

24.3.8 EEPROM Sizing (Word 0x12)

Provides indication on EEPROM size.

Bit	Name	Description
15:14	Signature	The *Signature* field indicates to the controller that there is a valid EEPROM present. If the signature field is 01b, EEPROM read is performed, otherwise the other bits in this word are ignored, no further EEPROM read is performed, and default values are used for the configuration space IDs.
13:10	EEPROM Size	These bits indicate the EEPROM's actual size. Mapped to EEC[14:11]. 0000b = Reserved 0001b = Reserved 0010b = Reserved 0011b = Reserved 0100b = Reserved 0101b = Reserved 0110b = Reserved 0111b = 16 KB 1000b = 32 KB 1001b = Reserved 1001b = Reserved 1011b = Reserved See Section 28.3.1.1
9:0	Reserved	Reserved

24.3.9 Initialization Control 4 (LAN Base Address + Offset 0x13)

These words control general initialization values of LAN 0, LAN 1, LAN 2, and LAN 3 ports.

Bit	Name	Default Value in EEPROM	Description
15:12	TXPbsize	0x8	Transmit internal buffer size: 0x0 - 20 KB 0x1 - 40 KB 0x2 - 80 KB 0x3 - 1 KB 0x4 - 2 KB 0x5 - 4 KB 0x6 - 8 KB 0x7 - 16 KB 0x8 - 19 KB 0x9 - 38 KB 0xA - 76 KB 0xB:0XF reserved. Note: When 4 ports are enabled maximum buffer size is 20KB. When 2 ports are enabled maximum buffer size is 40KB. When only a single port is enabled maximum buffer size is 80KB. Sets value of ITPBS.TXPbsize. See Section 28.2.
11:8	RXPbsize	0x8	Receive internal buffer size: 0x0 - 36 KB 0x1 - 72 KB 0x2 - 144 KB 0x3 - 1 KB 0x4 - 2 KB 0x5 - 4 KB 0x6 - 8 KB 0x7 - 16 KB 0x8 - 35 KB 0x9 - 70 KB 0xA - 140 KB Note: When 4 ports are enabled maximum buffer size is 36 KB. When 2 ports are enabled maximum buffer size is 72 KB. When only a single port is enabled maximum buffer size is 144 KB. Sets value of *IRPBS.RXPbsize*. See Section 28.2.
7	Reserved	0b	Reserved
6	LPLU	1b	Low Power Link Up Enables a decrease in link speed in non-D0a states when power policy and power management states dictate it.
5:1	Phy_Add	0x00 0x01 0x02 0x03	Phy Address. Value loaded to MDICNFG.PHY_ADD field.
0	DEV_RST_EN	1b	Enable software reset (*CTRL.DEV_RST*) generation to all ports (See Section 23.3).

24.3.10 PHY_RST_CONTROL (LAN Base Address + Offset 0x16)

These words control Phy reset for LAN0, 1, 2, 3.

Bit	Name	Default Value in EEPROM	Description
15:4	Reserved	0x0	Reserved
3	PHY_Rst_Cmpl_Prog_Done	0x1	This bit loads CGB_PHY_RST_CMPL_DLY CSR bit[3] when EEPROM load is done.
2:0	PHY_Rst_Cmpl_Dly	0x3	This field loads the initial value to CGB_PHY_RST_CMPL_DLY CSR bit[2:0] when EEPROM load is done.

24.3.11 Device Rev ID (Word 0x1E)

Bit	Name	Default Value in EEPROM	Description
15:8	Reserved	0x0	Reserved
7:0	DEVREVID	0x0	Device Revision ID The actual device revision ID is the EEPROM value XORed with the hardware default of Rev ID. For the Controller A0, the default value is zero.

24.3.12 LED 0 Configuration Defaults (LAN Base Address + Offset 0x1F)

These EEPROM words specify the hardware defaults for the LEDCTL register fields controlling the LED0 (LINK_UP) output behaviors. Words control LEDs behavior of LAN 0, LAN 1, LAN 2, and LAN 3 ports.

Bit	Name	Default Value in EEPROM	Description
3:0	LED0 Mode	0010b	Initial value of the *LED0_MODE* field specifying what event/state/pattern is displayed on LED0 (LINK_UP) output. A value of 0010b (0x2) indicates the LINK_UP state. See Section 28.1.4.9.
4	Reserved	0b	Reserved. Set to 0b.
5	Global Blink Mode	0b	Global Blink Mode 0b = Blink at 200 ms on and 200ms off. 1b = Blink at 83 ms on and 83 ms off. See Section 28.1.4.9.
6	LED0 Invert	0b	Initial value of *LED0_IVRT* field. 0b = Active-low output. See Section 28.1.4.9.
7	LED0 Blink	0b	Initial value of *LED0_BLINK* field. 0b = Non-blinking. See Section 28.1.4.9.
15:8	Reserved	0x0	Reserved

A value of 0x0002 is used to configure default hardware LED behavior (LED0=LINK_UP).

24.3.13 Software Defined Pins Control (LAN Base Address + Offset 0x20)

These words at offset 0x20 from start of relevant EEPROM section are used to configure initial settings of software defined pins (SDPs) for LAN 0, LAN 1, LAN 2 and LAN 3.

Bit	Name	Default Value in EEPROM	Description
15:12	Reserved	0x0	Reserved
11	LAN_DIS	0b	LAN Disable In LAN ports 0, 1, 2 and 3, when set to 1b, the appropriate LAN is disabled (both PCIe* function and LAN access for manageability are disabled). ***Note:*** • If LAN_DIS is set to 1b for a port then PHY CFG pointer (offset 0x50) must not have a PHY write command for that port set. See Section 24.6, "PHY Configuration Structure" on page 1077 • If all LAN ports are disabled (with LAN_DIS or LAN_PCI_DIS), then there is no way to update the EEPROM from an application running on the IA.
10	LAN_PCI_DIS	0b	LAN PCIe* Function Disable In LAN ports 0, 1, 2 and 3, when set to 1b, the appropriate LAN PCI function is disabled. For example, in the case where the LAN is functional for manageability operation but is not connected to the host through the PCIe* interface. Reflected in EEDIAG (See Section 28.3.1.3). ***Note:*** • If all LAN ports are disabled (with LAN_DIS or LAN_PCI_DIS), then there is no way to update the EEPROM from an application running on the IA.
9	SDPDIR[1]	0b	SDP1 Pin – Initial Direction This bit configures the initial hardware value of the *SDP1_IODIR* bit in the Device Control (CTRL) register following power up. See Section 28.1.4.2 .
8	SDPDIR[0]	0b	SDP0 Pin – Initial Direction This bit configures the initial hardware value of the *SDP0_IODIR* bit in the Device Control (CTRL) register following power up. See Section 28.1.4.2 .
7:6	Reserved	00b	Reserved
5	WD_SDP0	0b	When set, SDP[0] is used as a watchdog timeout indication. When reset, it is used as an SDP (as defined in bits 8 and 0). See Section 28.1.4.2 .
4:3	Reserved	00b	Reserved
2	D3COLD_WAKEUP_ADVEN	1b	Configures the initial hardware default value of the *ADVD3WUC* bit in the Device Control (CTRL) register following power up. See Section 28.1.4.2 .
1	SDPVAL[1]	0b	SDP1 Pin – Initial Output Value This bit configures the initial power-on value output on SDP1 (when configured as an output) by configuring the initial hardware value of the *SDP1_DATA* bit in the Device Control (CTRL) register after power up. See See Section 28.1.4.2 .
0	SDPVAL[0]	0b	SDP0 Pin – Initial Output Value This bit configures the initial power-on value output on SDP0 (when configured as an output) by configuring the initial hardware value of the *SDP0_DATA* bit in the Device Control (CTRL) register after power up. See Section 28.1.4.2 .

24.3.14 CGB Functions Control (Word 0x21)

Bits	Name	Default Value in EEPROM	Description
15:11	Reserved	0x0	Reserved
10	BAR32	0x1	Bit (loaded to the BARCTRL register) preserves the legacy 32 bit BAR mode when BAR32 is set. When cleared to 0b 64 bit BAR addressing mode is selected. Note: If PREFBAR is set the BAR32 bit should always be 0 (64 bit BAR addressing mode). Refer to bits[29:20] of Section 29.3.2, "Base Address Registers (0x10...0x27; R/W)".
9	PREFBAR	0x0	0 = BARs are marked as non prefetchable 1 = BARs are marked as prefetchable Note: PREFBAR should be set when 64 bit BARs are used. Bit should not be set in 32 bit BAR addressing mode. Refer to bits[29:20] of Section 29.3.2, "Base Address Registers (0x10...0x27; R/W)".
8:0	Reserved	0x0	Reserved

24.3.15 Initialization Control 3 (LAN Base Address + Offset 0x24)

These words control general initialization values of LAN 0, LAN 1, LAN 2 and LAN 3 ports.

Bit	Name	Default Value in EEPROM	Description
15	SerDes Energy Source	0b	SerDes Energy Source Detection When set to 0b, source is internal SerDes Rx circuitry for electrical idle or link-up indication. When set to 1b, source is external SRDS_[n]_SIG_DET signal for electrical idle or Link-up indication. This bit also indicates the source of the signal detect while establishing a link in SerDes mode. This bit sets the default value of the *CONNSW.ENRGSRC* bit.
14	2 wires SFP Enable	0b	2 wires SFP interface *enable* - bit is used to enable interfacing an external PHY either VIA the MDIO or I^2C interface 0b = Disabled. When disabled, the 2 wires I/F pads are isolated. 1b = Enabled. Used to set the default value of *CTRL_EXT.I2C Enabled*. See Section 28.1.4.4 . ***Note:*** Common MDIO is selected per LAN by setting Com_MDIO <bit 3> of Initialization Control 3 - If Common MDIO mode this bit must be set to 1b for LAN 0 - If LAN 1 is in Common MDIO mode set this bit for LAN 1 to 0b - If LAN 2 is in Common MDIO mode set this bit for LAN 2 to 0b - If LAN 3 is in Common MDIO mode set this bit for LAN 3 to 0b See note in Com_MDIO <bit 3> and MDIO_I2C <bit 2> of Initialization Control 3
13	ILOS	0b	Default setting for the loss-of-signal polarity bit (CTRL[7]). See Section 28.1.4.2
12:11		0x0	Reserved
10	APM Enable	0b	Initial value of *Advanced Power Management Wake Up Enable* bit in the Wake Up Control (WUC.APME) register. Mapped to CTRL[6] and to WUC[0]. See Section 28.1.4.2 and See Section 28.20.1.1 .
9	Reserved	0b	Reserved
8	ACBYP	0b	Bypass on-chip AC coupling in RX input buffers ACBYP = 0 -Normal mode; on-chip AC coupling present. ACBYP = 1 - On-chip AC coupling bypassed.
7	reserved	1b	reserved.
6	reserved	0b	reserved.
5:4	Link Mode	00b	Initial value of *Link Mode* bits of the Extended Device Control (CTRL_EXT.LINK_MODE) register, specifying which link interface and protocol is used by the MAC. 00b = Reserved Note: This is the lowest power mode. This is default for the PCH to keep MAC in lowest power mode by default. 01b = MAC and SerDes I/F operate in 1000BASE-KX mode. 10b = MAC and SerDes operate in SGMII mode. 11b = MAC and SerDes I/F operate in SerDes (1000BASE-BX) mode. See Section 28.1.4.4, "Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W)".

Bit	Name	Default Value in EEPROM	Description
3	Com_MDIO	0b	When interfacing an external SGMII phy or SerDes, bit defines if MDIO access is routed to a shared MDIO port on LAN 0, to support multi port external PHYs or to the dedicated per function MDIO port. 0b - MDIO access routed to the LAN port's MDIO interface. 1b - MDIO accesses on this LAN port routed to LAN port 0 MDIO interface. Note: If Common MDIO <this bit> is set for LAN 1, LAN 2 or LAN 3 then the following bits must be set to 1b for LAN 0 - This bit Com_MDIO <bit 3> - 2 wire SFP Enable <bit 14> Initialization Control 3 - MDIO_I2C <bit 2> Initialization Control 3 Used to set the default value of MDICNFG.Com_MDIO bit. See Section 28.1.4.6.
2	MDIO_I2C		MDIO/I2C Interface Selection: 0b = I2C 1b = MDIO Note: This bit must be set to 1b for each LAN that has Com_MDIO <bit 3> set. See note in Com_MDIO <bit 3>.
1	Ext_VLAN	0b	Sets the default for CTRL_EXT[26] bit. Indicates that additional VLAN is expected in this system. See Section 28.1.4.4 .
0[1]	Keep_PHY_Link_Up_En	0b	Enables *No PHY Reset* when the Baseboard Management Controller (BMC) indicates that the PHY should be kept on. When asserted, this bit prevents the PHY reset signal and the power changes reflected to the PHY according to the *MANC.Keep_PHY_Link_Up* value. Note: This EEPROM bit should be set to the same value for all LAN ports.

1. BMC uses this bit to keep MAC interface alive even if PCIe* function is off.

24.3.16 PCIe* Control 3 (Word 0x29)

This word is used for programming PCIe* functionality.

Bits	Name	Default Value in EEPROM	Description
15:6	Reserved	0x0	Reserved
5	Wake_pin_enable	0b	Enables the use of the wake pin for a PME event in all non L2 power states. WAKE_N pin will be asserted in the following condition: 1) wake enable from LAN port filter state machine 2) no PCIe* power i.e pe_rst_n asserted If the pe_rst_n is deasserted then this bit from EEPROM can disable the above gating for example wake pin will be always asserted when LAN port filter indicates pme.
4:0	Reserved	0x0	Reserved

24.3.17 Watchdog Configuration (Word 0x2E)

Bit	Name	Default Value in EEPROM	Description
15	Watchdog Enable	0b	Enable watchdog interrupt. See Section 28.15.1.1.
14:11	Watchdog Timeout	0x2	Watchdog timeout period (in seconds). See Section 28.15.1.1.
10:0	Reserved	0x0	Reserved

24.3.18 VPD Pointer (Word 0x2F)

This word points to the Vital Product Data (VPD) structure. This structure is available for the NIC vendor to store it's own data. A value of 0xFFFF indicates that the structure is not available.

Bit	Name	Description
15:0	VPD offset	Offset to VPD structure in words. Bit 15 must be set to 0b (VPD area must be in the first 32 Kbytes of the EEPROM).

24.4 CSR Auto Configuration Pointer (LAN Base Address + Offset 0x17)

Words points to the CSR auto configuration structures of LAN 0, LAN 1, LAN 2 and LAN3. Sections are loaded during HW auto-load as described in Section 22.3.1.3. If no CSR autoload is required for the specific LAN port, the word shall be set to 0xFFFF.

The CSR Auto Configuration structure format is listed in the Table 24-5.

For LAN 0, 1, 2, or 3 this structure may contain Generic LANx CSR load values.

Table 24-5. CSR Auto Configuration Structure Format

Offset	High Byte[15:8]	Low Byte[7:0]	Section
0x0	Section Length = 3*n (n – number of CSRs to configure)		Section 24.4.1
0x1	Block CRC8		Section 24.4.2
0x2	CSR Address		Section 24.4.3
0x3	Data LSB		Section 24.4.4
0x4	Data MSB		Section 24.4.5
	…		
3*n - 1	CSR Address		Section 24.4.3
3*n	Data LSB		Section 24.4.4
3*n + 1	Data MSB		Section 24.4.5

The CSR Auto Configuration structure format is listed in the following tables.

24.4.1 CSR Configuration Section Length - Offset 0x0

The section length word contains the length of the section in words. Section length count does not include the section length word and Block CRC8 word.

Bits	Name	Default	Description
15:0	Section_length		Section length in words.

24.4.2 Block CRC8 (Offset 0x1)

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.4.3 CSR Address - Offset 0x2

The CSR Address field is used to access one of the MMIO Registers listed in the "Programming Interface" chapter. However, in this structure the CSR address must only specify the aligned DWORD address of the given register, for example, only bits [15:2] of the register address as specified in the Programming Interface chapter must be used.

Bits	Name	Default	Description
15:0	CSR_ADDR		CSR Address in Double Words (4 bytes)

24.4.4 CSR Data LSB - Offset 0x3

Bits	Name	Default	Description
15:0	CSR_Data_LSB		CSR Data LSB

24.4.5 CSR Data MSB - Offset 0x4

Bits	Name	Default	Description
15:0	CSR_Data_MSB		CSR Data MSB

24.5 CSR Auto Configuration Power-Up Pointer (LAN Base Address + Offset 0x27)

Words points to the CSR auto configuration Power-Up structures of LAN 0, LAN 1, LAN 2 and LAN3 that are read only following power-up. Sections are loaded during HW auto-load as described in Section 22.3.1.3. If no CSR autoload is required for the specific LAN port, the word shall be set to 0xFFFF.

The CSR Auto Configuration Power-Up structure format is listed Table 24-6.

For LAN 0 the structure may contain three sub-sections in the following order: PCIe* MPHY, LAN 0 MPHY, and Generic LAN 0 CSR load values.

For LAN 1, 2, or 3 the structure may contain 2 sub-sections in the following order: LANx MPHY, and Generic LANx CSR load values.

There is only a Section Length and Block CRC8 per structure block regardless of the number of sub-sections in each block. The hardware simply writes the MSB/LSB Data pair to the CSR Address provided.

Table 24-6. CSR Auto Configuration Power-Up Structure Format

Offset	High Byte[15:8]	Low Byte[7:0]	Section
0x0	Section Length = 3*n (n – number of CSRs to configure)		Section 24.5.1
0x1	Block CRC8		Section 24.5.2
0x2	CSR Address		Section 24.5.3
0x3	Data LSB		Section 24.5.4
0x4	Data MSB		Section 24.5.5
	…		
3*n - 1	CSR Address		Section 24.5.3
3*n	Data LSB		Section 24.5.4
3*n + 1	Data MSB		Section 24.5.5

24.5.1 CSR Configuration Power-Up Section Length - Offset 0x0

The section length word contains the length of the section in words. Section length count does not include the section length word and Block CRC8 word.

Bits	Name	Default	Description
15:0	Section_length		Section length in words.

24.5.2 Block CRC8 (Offset 0x1)

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.5.3 CSR Address - Offset 0x2

The CSR Address field is used to access one of the MMIO Registers listed in the "Programming Interface" chapter. However, in this structure the CSR address must only specify the aligned DWORD address of the given register, for example, only bits [15:2] of the register address as specified in the Programming Interface chapter must be used.

Bits	Name	Default	Description
15:0	CSR_ADDR		CSR Address in Double Words (4 bytes)

24.5.4 CSR Data LSB - Offset 0x3

Bits	Name	Default	Description
15:0	CSR_Data_LSB		CSR Data LSB

24.5.5 CSR Data MSB - Offset 0x4

Bits	Name	Default	Description
15:0	CSR_Data_MSB		CSR Data MSB

24.5.6 EICT Information Structure

The EEPROM space prior to the beginning of the CSR Auto Configuration Power-Up Structure for each LANx when be preserved to accommodate the EICT revision history. Table 24-7 describes the structure entries.

Table 24-7. EICT Information Structure Format

Variable Name	Description	Size
U8 solution[8]	Read from mphy file - bin min trace length in inches, bin max trace length in inches.	8 Bytes
U8 type[8]	File type PCIe*, SerD SFP, SerD Bkp, SGMI, SGMI Bkp.	8 Bytes
U16 version	Version number between 00 – 99 read from file.	2 Bytes
U16 size	Size in words, calculated by EICT.	2 Bytes
U16 strucVer	This would be set/used by EICT to version this structure. For this structure the value would be 0x1 set by EICT.	2 Bytes
U16 Signature	Always 0x55AA set by EICT.	2 Bytes

24.6 PHY Configuration Structure

This section describes the PHY auto configuration structure used to configure PHY related circuitry. The programming in this section is applied after each PHY reset. After the registers in this section are written to the PHY, the relevant *EEMNGCTL.CFG_DONE* bit is set.

The PHY Configuration Pointer (Word 0x50) points to the start (offset 0x0) of this type of structure, to configure PHY registers (External PHYs). If pointer is 0xFFFF then no structure exists. Structure is loaded during HW EEPROM auto-load.

Table 24-8. PHYAuto Configuration Structure Format

Offset	High Byte[15:8]	Low Byte[7:0]	Section
0x0	Section Length = 2*n (n – number of registers to configure)		
0x1	Block CRC8		Table 24-10
0x2	PHY number and PHY address		Table 24-11
0x3	PHY data (MDIC[15:0] or I2CCMD[15:0])		Table 24-12
	…		
2*n	PHY number and PHY address		Table 24-11
2*n + 1	PHY data (MDIC[15:0] or I2CCMD[15:0])		Table 24-12

24.6.1 PHY Configuration Section Length - Offset 0x0

The section length word contains the length of the section in words. Section length count does not include the section length word and Block CRC8 word.

Table 24-9. HY Configuration Section Length

Bits	Name	Default	Description
15	Reserved		
14:0	Section_length		Section length in words.

24.6.2 Block CRC8 (Offset 0x1)

Table 24-10. Block CRC8 Structure

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.6.3 PHY Number and PHY Address - (Offset 2*n; [n = 1... Section Length])

Table 24-11. PHY Number and PHY Address

Bits	Name	Default	Description
15:12	Reserved		Reserved
11	Apply to port 3		If set, apply to programming when the PHY of port three is reset. Note: Must be disabled/cleared when Port 3 is disabled via LAN_DIS.
10	Apply to port 2		If set, apply to programming when the PHY of port two is reset. Note: Must be disabled/cleared when Port 2 is disabled via LAN_DIS.
9	Apply to port 1		If set, apply to programming when the PHY of port one is reset. Note: Must be disabled/cleared when Port 1 is disabled via LAN_DIS.
8	Apply to port 0		If set, apply to programming when the PHY of port zero is reset. Note: Must be disabled/cleared when Port 0 is disabled via LAN_DIS.
7:0	PHY address		PHY address to which the data is written

24.6.4 PHY Data (Offset 2*n + 1; [n = 1... Section Length])

Table 24-12. PHY Data

Bits	Name	Default	Description
15:0	Reg_Data		MDIC[15:0]/I2CCMD[15:0] value (Data).

24.7 Pointers and Control Words Used By Firmware

Word 0x51 is used to point to the load Firmware patch structure. Word 0x11 in each LAN port EEPROM section are used to point to structures specific to Pass through operation. Words 0x54 and 0x23 control some aspects of the Firmware functionality.

A value of 0xFFFF for a pointer indicates that the relevant structure is not present in the EEPROM.

24.7.1 Pass Through LAN Configuration Pointer (LAN Base Address + Offset 0x11)

Bit	Name	Description
15:0	Pointer	Pointer to the PT LAN Configuration Structure. See Section 24.9 for details of the structure.

24.7.2 PHY Configuration Pointer (Word 0x50)

Bit	Name	Description
15:0	Pointer	Pointer to PHY configuration structure. A value of 0xFFFF means the pointer is invalid.

24.7.3 Firmware Patch Pointer (Word 0x51)

Bit	Name	Description
15:0	Pointer	Pointer to loader patch structure. See Section 24.8 for details of the structure.

24.7.4 Sideband Configuration Pointer (Word 0x57)

Bit	Name	Description
15:0	Pointer	Pointer to the Sideband configuration pointer structure. See Section 24.10 for details of the structure.

24.7.5 Manageability Capability/Manageability Enable (Word 0x54)

Bit	Name	Description
15	Reserved	Reserved
14	Redirection Sideband Interface	0b = SMBus. (For PCH, it needs to be set to 0)
13:12	Reserved	Reserved.
11	MCSR_TO_RETRY	0b - On CSR access Timeout return failed access to BMC. 1b - On CSR access Timeout retry access internally and don't return failed access. Default value - 1b
10:8	Manageability Mode	0x0 = None. 0x1 = Reserved. 0x2 = Pass Through (PT) mode. 0x3 = Reserved. 0x4 = Host interface enable only. 0x5:0x7 = Reserved.
7	Port3 Manageability Capable	0b = Not capable 1b = Bit 3 is applicable to port 3. Note: Must be set if bit 3 is set.
6	Port2 Manageability Capable	0b = Not capable 1b = Bit 3 is applicable to port 2. Note: Must be set if bit 3 is set.
5	Port1 Manageability Capable	0b = Not capable 1b = Bit 3 is applicable to port 1. Note: Must be set if bit 3 is set.
4	Port0 Manageability Capable	0b = Not capable 1b = Bit 3 is applicable to port 0. Note: Must be set if bit 3 is set.
3	Pass Through Capable	0b = Disable. 1b = Enable. Must be set if bits[10:8] (Manageability Mode) = 0x2 (PT).
2:0	Reserved	Reserved

24.7.6 Management HW Config Control (Word 0x23)

This word contains bits that configure special firmware behavior.

Bit	Name	Description
15	LAN3_FTCO_RST_DIS	LAN3 force TCO reset disable (1b disable; 0b enable).
14	LAN2_FTCO_RST_DIS	LAN2 force TCO reset disable (1b disable; 0b enable).
13	LAN1_FTCO_RST_DIS	LAN1 force TCO reset disable (1b disable; 0b enable).
12	LAN0_FTCO_RST_DIS	LAN0 force TCO reset disable (1b disable; 0b enable).
11	Reserved	Reserved
10	PARITY_ERR_RST_EN	When set enables reset of Management logic and generation of internal Firmware reset as a result of Parity Error detected in Management memories.
9	Enable All Phys in D3	0 - Only ports activated for BMC activity will stay active in D3: In SMBUS mode – according to EEPROM port enable bit. 1- All phys will stay active in D3.
8:4	Reserved	Reserved
3	LAN3_FTCO_ISOL_DIS	LAN3 force TCO Isolate disable (1b disable; 0b enable).
2	LAN2_FTCO_ISOL_DIS	LAN2 force TCO Isolate disable (1b disable; 0b enable).
1	LAN1_FTCO_ISOL_DIS	LAN1 force TCO Isolate disable (1b disable; 0b enable).
0	LAN0_FTCO_ISOL_DIS	LAN0 force TCO Isolate disable (1b disable; 0b enable).

24.8 Firmware Patch Structure

This structure is used for all FW patches. Firmware patch Pointer (Word 0x51) points to the start (offset 0x0) of this kind of structure. If pointer is 0xFFFF then no structure exists. Structure is loaded during HW EEPROM auto-load as described in Chapter 22.0, "GbE Interconnects".

24.8.1 Firmware Patch Data Size (Offset 0x0)

The Data Size word contains the length of the section in words. Data Size count does not include the section length word and Block CRC8 word.

Bit	Name	Description
15:0	Data Size (Words)	

24.8.2 Block CRC8 (Offset 0x1)

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.8.3 Patch Ram Address Word (Offset 0x2)

Bit	Name	Description
15:0	Ram address	Ram Address to load patch.

24.8.4 Patch Version 1 Word (Offset 0x3)

Bit	Name	Description
15:8	Patch Generation Hour	
7:0	Patch Generation Minutes	

24.8.5 Patch Version 2 Word (Offset 0x4)

Bit	Name	Description
15:8	Patch Generation Month	
7:0	Patch Generation Day	

24.8.6 Patch Version 3 Word (Offset 0x5)

Bit	Name	Description
15:8	Patch Silicon Version Compatibility	0x00 = A0. 0x01 = A1. 0x10 = B0. 0x11 = B1.
7:0	Patch Generation Year	

24.8.7 Patch Version 4 Word (Offset 0x6)

Bit	Name	Description
15:8	Patch Major Number	
7:0	Patch Minor Number	

24.8.8 Patch Data Words (Offset 0x7, Block Length)

Bit	Name	Description
15:0	Patch Firmware Data	

24.9 PT LAN Configuration Structure

The Pass Through LAN Configuration Pointer (LAN Base Address + Offset 0x11) points to the start (offset 0x0) of this type of structure, to configure manageability filters. If pointer is 0x FFF then no structure exists. Structure is loaded during HW EEPROM auto-load.

When interface to BMC is via SMBus the PT LAN Configuration Structure must be used to configure the required filters. The PT LAN Configuration Structure format is listed in the following tables.

Table 24-13. PT LAN Configuration Structure Format

Offset	High Byte[15:8]	Low Byte[7:0]	Section
0x0	Section Length = 3*n (n – number of CSRs to configure)		Section 24.9.1
0x1	Block CRC8		Section 24.9.2
0x2	CSR Address		Section 24.9.3
0x3	Data LSB		Section 24.9.4
0x4	Data MSB		Section 24.9.5
	…		
3*n - 1	CSR Address		Section 24.9.3
3*n	Data LSB		Section 24.9.4
3*n + 1	Data MSB		Section 24.9.5

24.9.1 PT LAN Configuration Structure Section Length - Offset 0x0

The section length word contains the length of the section in words. Section length count does not include the section length word and Block CRC8 word.

Bits	Name	Default	Description
15	Reserved		
14:0	Section_length		Section length in words.

24.9.2 Block CRC8 (Offset 0x1)

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.9.3 CSR Address - (Offset 2*n; [n = 1... Section Length])

Bits	Name	Default	Description
15	Reserved		
14:0	CSR_ADDR		CSR Address in Double Words (4 bytes)

24.9.4 CSR Data LSB - (Offset 0x1 + 2*n; [n = 1... Section Length])

Bits	Name	Default	Description
15:0	CSR_Data_LSB		CSR Data LSB

24.9.5 CSR Data MSB - (Offset 0x2 + 2*n; [n = 1... Section Length])

Bits	Name	Default	Description
15:0	CSR_Data_MSB		CSR Data MSB

24.9.6 Manageability Filters

The following table lists registers that can be programmed via the PT LAN Configuration structure.

Name	Description	Section
MAVTV	Management VLAN TAG Value	Section 28.21.1.1, "Management VLAN TAG Value—MAVTV [0:3][0:7] (0x5010 +4*n [n=0...7]; RW)"
MFUTP	Management Flex UDP/TCP Ports	Section 28.21.1.2, "Management Flex UDP/TCP Ports—MFUTP[0:3][0:3] (0x5030 + 4*n [n=0...3]; RW)"
METF	Management Ethernet Type Filters	Section 28.21.1.3, "Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW)"
MNGONLY	Management Only Traffic Register	Section 28.21.1.4, "Management Control Register—MANC [0:3] (0x5820; RW)"
MDEF	Manageability Decision Filters	Section 28.21.1.6, "Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW)"
MDEF_EXT	Manageability Decision Filters Extension	Section 28.21.1.7, "Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW)"
MIPAF	Manageability IP Address Filter registers (IPv4 or IPV6) Can be used to filter IPV4 Address of ARP packets.	Section 28.21.1.8, "Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW)"
MMAL	Manageability MAC Address Low Registers	Section 28.21.1.9, "Manageability MAC Address Low—MMAL [0:3] (0x5910 + 8*n [n= 0...1]; RW)"
MMAH	Manageability MAC Address High Registers	Section 28.21.1.10, "Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0...1]; RW)"
FTFT	Flexible TCO Filter Table registers	Section 28.21.1.11, "Flexible TCO Filter Table Registers—FTFT [0:3] (0x9400-0x94FC; RW)"

24.10 Sideband Configuration Structure

The Sideband Configuration Pointer (Word 0x57) points to the start (offset 0x0) of this structure. If pointer is 0xFFFF then no structure exists.

24.10.1 Section Length (Offset 0x0)

The section length word contains the length of the section in words. Section length count does not include the section length word and Block CRC8 word.

Bits	Name	Default	Description
15:0	Section_length	0	Section length in words.

24.10.2 Block CRC8 (Offset 0x1)

Bit	Name	Description
15:8	Reserved	
7:0	CRC8	

24.10.3 SMBus Max Fragment Size (Offset 0x2)

Bit	Name	Description
15:8	Reserved	Reserved
7:0	SMBus Max Fragment Size (Bytes)	Maximum SMBus Fragment Size. Value should be in the 32 to 240 Byte range. Note: Fragment size should be a multiple of 4 + 1 (for example, 33, 37...).

24.10.4 SMBus Notification Timeout (Offset 0x3)

Bit	Name	Description
15:0	SMBus Notification Timeout (ms)	Timeout until the discarding of a packet not read by the external BMC completes. 0b - No discard.

24.10.5 SMBus Slave Address 0 1 (Offset 0x4)

Bit	Name	Description
15:9	SMBus 1 Slave Address	SMBus Slave Address for port 1.
8	Reserved	Reserved.
7:1	SMBus 0 Slave Address	SMBus Slave Address for port 0.
0	Reserved	Reserved.

24.10.6 SMBus Slave Address 2 3 (Offset 0x5)

Bit	Name	Description
15:9	SMBus 3 Slave Address	SMBus Slave Address for port 3.
8	Reserved	Reserved.
7:1	SMBus 2 Slave Address	SMBus Slave Address for port 2.
0	Reserved	Reserved.

24.10.7 Reserved (Offset 0x6)

Bit	Name	Description
15:0	Reserved	Reserved

24.10.8 LAN Receive Enable 1 (Offset 0x7)

Bit	Name	Description
15:9	Receive Enable Byte 12	BMC SMBus slave address.
8	Reserved	Reserved
7	Reserved	Reserved
6	Reserved	Reserved
5:4	Notification Method	00b = SMB alert. 01b = Asynchronous notify. 10b = Direct receive. 11b = Reserved.
3	Enable ARP Response	0 - Disable ARP Response 1 - Enable ARP Response
2	Enable Status Reporting	
1:0	Reserved	Reserved

24.10.9 LAN Receive Enable 2 (Offset 0x8)

Bit	Name	Description
15:8	Receive Enable Byte 14	Alert value.
7:0	Receive Enable Byte 13	Interface value.

24.10.10 SMBus Flags (Offset 0x9)

Bit	Name	Description
15:6	Reserved	Reserved
5	SMBus Block Read Command	0b = Block read command is C0. 1b = Block read command is D0.
4:3	Reserved	Reserved
2	Disable SMBus ARP Functionality	
1	SMBus ARP PEC	
0	Reserved	

24.10.11 LAN Receive Enable 3 (Offset 0xA)

Bit	Name	Description
15:4	Reserved	Reserved
3	Enable BMC Dedicated MAC port 3	Configure BMC dedicated MAC Address on Port 3. 0b = The PCH will share MAC address for MNG traffic with host MAC address on Port 3. Host MAC address is specified in EEPROM words located at LAN Base Address + offset 0x0-0x2 per port. 1b = The PCH will use the BMC dedicated MAC address on port 3 (MAC address (MMAL/H)as filter for incoming receive packets. Note: MMAL/H registers can be programmed from the EEPROM using the *PT LAN Configuration structure*.
2	Enable BMC Dedicated MAC port 2	Configure BMC dedicated MAC Address on Port 2. 0b = The PCH will share MAC address for MNG traffic with host MAC address on Port 2. Host MAC address is specified in EEPROM words located at LAN Base Address + offset 0x0-0x2 per port . 1b = The PCH will use the BMC dedicated MAC address on port 2(MAC address (MMAL/H)as filter for incoming receive packets. Note: MMAL/H registers can be programmed from the EEPROM using the *PT LAN Configuration structure*.
1	Enable BMC Dedicated MAC port 1	Configure BMC dedicated MAC Address on Port 1. 0b = The PCH will share MAC address for MNG traffic with host MAC address on Port 1. Host MAC address is specified in EEPROM words located at LAN Base Address + offset 0x0-0x2 per port. 1b = The PCH will use the BMC dedicated MAC address on port 1(MAC address (MMAL/H) as filter for incoming receive packets. Note: MMAL/H registers can be programmed from the EEPROM using the *PT LAN Configuration structure*.
0	Enable BMC Dedicated MAC port 0	Configure BMC dedicated MAC Address on Port 0. 0b = The PCH will share MAC address for MNG traffic with host MAC address on Port 0. Host MAC address is specified in EEPROM words located at LAN Base Address + offset 0x0-0x2 per port. 1b = The PCH will use the BMC dedicated MAC address on port 3 (MAC address (MMAL/H) as filter for incoming receive packets. Note: MMAL/H registers can be programmed from the EEPROM using the *PT LAN Configuration structure*.

This value will be stored in the LSB Portion of the LAN0 Management Control (MANC) register.

Bit	Name	Description
15	TCO_Isolate	TCO Isolate command. Note: Value placed in this field is not written to MANC register.
14	FW_RESET	FW Reset occurred. Note: Value placed in this field is not written to MANC register.
13:0	Reserved	Reserved.

This value will be stored in the MSB Portion of the LAN0 Management Control (MANC) register.

Bit	Name	Description
15:11	Reserved	Reserved.
10	NET_TYPE	NET TYPE: 0b = pass only un-tagged packets. 1b = pass only VLAN tagged packets. Valid only if FIXED_NET_TYPE is set.
9	FIXED_NET_TYPE	Fixed net type: If set, only packets matching the net type defined by the NET_TYPE field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine.
8	EN_IPv4_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.
7	EN_XSUM_FILTER	Enable checksum filtering to MNG When this bit is set, only packets that pass L3, L4 checksum are sent to the MNG block.
6:4	Reserved	Reserved should be 0
2	KEEP_PHY_LINK_UP	Block PHY reset and power state changes. When this bit is set the PHY reset and power state changes do not reach to the PHY (PHY is not reset), This bit can not be written to, unless the *Keep_PHY_Link_Up_En* EEPROM bit is set.
1	RCV_TCO_EN	Enable BMC to network and network to BMC traffic 0b - The BMC can not communicate with the network. 1b - The BMC can communicate with the network When cleared the BMC traffic is not forwarded to the network and the network traffic is not forwarded to the BMC even if the decision filters indicates it should. This bit does not impact the OS to BMC traffic.
0	TCO_RESET	TCO Reset occurred Note: Value placed in this field is not written to MANC register.

This value will be stored in the LSB Portion of the LAN1 Management Control (MANC) register.

Bit	Name	Description
15	TCO_Isolate	TCO Isolate command. Note: Value placed in this field is not written to MANC register.
14	FW_RESET	FW Reset occurred. Note: Value placed in this field is not written to MANC register.
13:0	Reserved	Reserved.

This value will be stored in the MSB Portion of the LAN1 Management Control (MANC) register.

Bit	Name	Description
15:11	Reserved	Reserved.
10	NET_TYPE	NET TYPE: 0b = pass only un-tagged packets. 1b = pass only VLAN tagged packets. Valid only if FIXED_NET_TYPE is set.
9	FIXED_NET_TYPE	Fixed net type: If set, only packets matching the net type defined by the NET_TYPE field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine.
8	EN_IPv4_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.
7	EN_XSUM_FILTER	Enable checksum filtering to MNG When this bit is set, only packets that pass L3, L4 checksum are sent to the MNG block.
6:4	Reserved	Reserved should be 0
2	KEEP_PHY_LINK_UP	Block PHY reset and power state changes. When this bit is set the PHY reset and power state changes do not reach to the PHY (PHY is not reset), This bit can not be written to, unless the *Keep_PHY_Link_Up_En* EEPROM bit is set.
1	RCV_TCO_EN	Enable BMC to network and network to BMC traffic The BMC can not communicate with the network. The BMC can communicate with the network When cleared the BMC traffic is not forwarded to the network and the network traffic is not forwarded to the BMC even if the decision filters indicates it should. This bit does not impact the OS to BMC traffic.
0	TCO_RESET	TCO Reset occurred Note: Value placed in this field is not written to MANC register.

This value will be stored in the LSB Portion of the LAN2 Management Control (MANC) register.

Bit	Name	Description
15	TCO_Isolate	TCO Isolate command. Note: Value placed in this field is not written to MANC register.
14	FW_RESET	FW Reset occurred. Note: Value placed in this field is not written to MANC register.
13:0	Reserved	Reserved.

This value will be stored in the MSB Portion of the LAN2 Management Control (MANC) register.

Bit	Name	Description
15:11	Reserved	Reserved.
10	NET_TYPE	NET TYPE: 0b = pass only un-tagged packets. 1b = pass only VLAN tagged packets. Valid only if FIXED_NET_TYPE is set.
9	FIXED_NET_TYPE	Fixed net type: If set, only packets matching the net type defined by the NET_TYPE field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine.

Bit	Name	Description
8	EN_IPv4_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.
7	EN_XSUM_FILTER	Enable checksum filtering to MNG When this bit is set, only packets that pass L3, L4 checksum are sent to the MNG block.
6:4	Reserved	Reserved should be 0
2	KEEP_PHY_LINK_UP	Block PHY reset and power state changes. When this bit is set the PHY reset and power state changes do not reach to the PHY (PHY is not reset), This bit can not be written to, unless the *Keep_PHY_Link_Up_En* EEPROM bit is set.
1	RCV_TCO_EN	Enable BMC to network and network to BMC traffic The BMC can not communicate with the network. The BMC can communicate with the network When cleared the BMC traffic is not forwarded to the network and the network traffic is not forwarded to the BMC even if the decision filters indicates it should. This bit does not impact the OS to BMC traffic.
0	TCO_RESET	TCO Reset occurred Note: Value placed in this field is not written to MANC register.

This value will be stored in the LSB Portion of the LAN3 Management Control (MANC) register.

Bit	Name	Description
15	TCO_Isolate	TCO Isolate command. Note: Value placed in this field is not written to MANC register.
14	FW_RESET	FW Reset occurred. Note: Value placed in this field is not written to MANC register.
13:0	Reserved	Reserved.

This value will be stored in the MSB Portion of the LAN3 Management Control (MANC) register.

Bit	Name	Description
15:11	Reserved	Reserved.
10	NET_TYPE	NET TYPE: 0b = pass only un-tagged packets. 1b = pass only VLAN tagged packets. Valid only if FIXED_NET_TYPE is set.
9	FIXED_NET_TYPE	Fixed net type: If set, only packets matching the net type defined by the NET_TYPE field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine.
8	EN_IPv4_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.
7	EN_XSUM_FILTER	Enable checksum filtering to MNG When this bit is set, only packets that pass L3, L4 checksum are sent to the MNG block.
6:4	Reserved	Reserved should be 0
2	KEEP_PHY_LINK_UP	This bit gates the ability for MNG to be able to modify bit 18 (KEEP_PHY_LINK_UP) in Management Control Register - MANC [0:3] (0x5820; RW). If this bit is set, MNG can modify bit 18 in MANCx, otherwise, bit 18 of MANCx cannot be modified by MNG.

Bit	Name	Description
1	RCV_TCO_EN	Enable BMC to network and network to BMC traffic The BMC can not communicate with the network. The BMC can communicate with the network When cleared the BMC traffic is not forwarded to the network and the network traffic is not forwarded to the BMC even if the decision filters indicates it should. This bit does not impact the OS to BMC traffic.
0	TCO_RESET	TCO Reset occurred Note: Value placed in this field is not written to MANC register.

24.11 Software Accessed Words

Words 0x03 to 0x07 in the EEPROM image are reserved for compatibility information. New bits within these fields will be defined as the need arises for determining software compatibility between various hardware revisions.

Words 0x8 and 0x09 are used to indicate the Printed Board Assembly (PBA) number and words 0x42 and 0x43 identifies the EEPROM image.

Words 0x30 to 0x3E have been reserved for configuration and version values to be used by PXE code. The only exception is word 0x37 used for Pointer to Alternate MAC address.

24.11.1 Compatibility (Word 0x03)

Bit	Description
15:12	Reserved (set to 0000b).
11	LOM/Not a LOM. 0b = NIC. 1b = LOM.
10	Server/Not a Server NIC. 0b = Client. 1b = Server.
9	Client/Not a Client NIC. 0b = Server. 1b = Client.
8	Retail/OEM. 0b = Retail. 1b = OEM.
7:6	Reserved (set to 00b).
5	Reserved (set to 1b).
4	SMBus Connected. 0b = Not connected. 1b = Connected.
3	Reserved (set to 0b).
2	Reserved (set to 0b).
1:0	Reserved (set to 00b)

24.11.2 Port Identification LED Blinking (Word 0x04)

Driver Software provides a method to identify an external port on a system through a command that causes the LED to blink. Based on the setting in word 0x4 the LED should blink between STATE1 and STATE2 when a port identification command is issued. STATE1 and STATE2 are software driver states between which software toggles and the table shown below indicates how the LED is driven by software within each state.

When word 0x4 is equal to 0xFFFF the blinking behavior revert to a default.

Bit	Description
15:12	Control for LED 3. 0001b = Default in STATE1 + Default in STATE2. 0010b = Default in STATE1 + LED is ON in STATE2. 0011b = Default in STATE1 + LED is OFF in STATE2. 0100b = LED is ON in STATE1 + Default in STATE2. 0101b = LED is ON in STATE1 + LED is ON in STATE2. 0110b = LED is ON in STATE1 + LED is OFF in STATE2. 0111b = LED is OFF in STATE1 + Default in STATE2. 1000b = LED is OFF in STATE1 + LED is ON in STATE2. 1001b = LED is OFF in STATE1 + LED is OFF in STATE2. Note: Default is the LED behavior in STATE1 or STATE2 based on the value indicated in LED Control Register (LEDCTL) bits[3:0].
11:8	Control for LED 2 – same encoding as for LED 3.
7:4	Control for LED 1 – same encoding as for LED 3.
3:0	Control for LED 0 – same encoding as for LED 3.

24.11.3 OEM Specific (Word 0x06, 0x07)

These words are available for OEM use.

24.11.4 EEPROM Image Revision (Word 0x05)

This word is valid only for device starter images and indicates the ID and version of the EEPROM image. This word is used for tracking the EEPROM image revision and does not have any effect on the hardware functionality.

Bit	Description
15:12	EEPROM major version.
11:4	EEPROM minor version.
3:0	EEPROM image ID.

24.11.5 PBA Number (Word 0x09)

Word 0x09 contains the pointer to the PBA structure.

Bit	Description
15:00	Pointer to PBA Structure

24.11.5.1 PBA Structure

The Printed Board Assembly (PBA) number that are stored in a multi-byte field. The purpose of this information is to allow customer support (or any user) to identify the exact revision level of a product. Network driver software should not rely on this field to identify the product or its capabilities.

Note: This PBA number is not related to the MSI-X Pending Bit Array (PBA).

24.11.6 PXE Main Setup Options (Word 0x30/0x34/0x38/0x3A)

The configuration of the PXE software is controlled by EEPROM on the adapter. The main setup options are stored in word 0x30 an word 0x34. These options are those that can be changed by the user via the Control-S setup menu. The default value for this word is 0x0100. The signature bits in Word 0x31 will be checked by software to determine whether the contents of this word are valid.

Word 0x30 is used to configure the LAN0 port, word 0x34 is used to configure the LAN1 port, word 0x38 is used to configure the LAN2 port, and word 0x3A is used to configure the LAN3 port.

These Words are only mapped in GbE0 space (Function 1).

Bit	Description
15:13	Reserved. Must be 0.
12:10	Bits 12-10 control forcing speed and duplex during driver operation. Valid values are: 000b - Auto negotiate 001b - 10Mbps Half Duplex 010b - 100Mbps Half Duplex 011b - Not valid (treated as 000b) 100b - Not valid 101b - 10Mbps Full Duplex 110b - 100Mbps Full Duplex 111b - Not valid ***Note:*** Only applicable for copper-based adapters. Default value is 000b. • 1000Mbps is an auto negotiated option.
9	Reserved
8	Display Setup Message. If the bit is set to 1, the "Press Control-S" message is displayed after the title message. Default value is 1.
7:6	Prompt Time. These bits control how long the "Press Control-S" setup prompt message is displayed, if enabled by bit 8 00 = 2 seconds (default) 01 = 3 seconds 10 = 5 seconds 11 = 0 seconds Note: The Ctrl-S message is not displayed if 0 seconds prompt time is selected.
5	Deprecated. Must be 0.

Bit	Description
4:3	Default Boot Selection. These bits select which device is the default boot device. These bits are only used if the agent detects that the BIOS does not support boot order selection or if the MODE field of word 31h is set to .Legacy mode 00 = Network boot, then local boot (default) 01 = Local boot, then network boot 10 = Network boot only 11 = Local boot only
2	Deprecated. Must be 0.
1:0	Protocol Select. These bits select the active boot protocol: 00 = PXE (default value) 01 = Reserved 10 = Reserved 11 = Reserved Only the default value of 00b should be initially programmed into the adapter; other values should only be set by configuration utilities.

24.11.7 PXE Configuration Customization Options (Word 0x31/0x35/0x39/0x3B)

Word 0x31 and word 0x35 of the EEPROM contains settings that can be programmed by an OEM or network administrator to customize the operation of the software. These settings cannot be changed from within the Control-S setup menu. The lower byte contains settings that would typically be configured by a network administrator using an external utility; these settings generally control which setup menu options are changeable. The upper byte is generally settings that would be used by an OEM to control the operation of the agent in a LOM environment, although there is nothing in the agent to prevent their use on a NIC implementation. The default value for this word is 0x4000.

Word 0x31 is used to configure the primary port and word 0x35 is used to configure the secondary port, word 0x39 is used to configure the LAN2 port, and word 0x3B is used to configure the LAN3 port.

These Words are only mapped in GbE0 space (Function 1).

Bit	Description
15:14	Signature. Must be set to 01 to indicate that this word has been programmed by the agent or other configuration software.
13:12	Reserved - must be 0
11	Selects Continuous Retry operation. If this bit is set, IBA will NOT transfer control back to the BIOS if it fails to boot due to a network error (such as failure to receive DHCP replies). Instead, it will restart the PXE boot process again. If this bit is set, the only way to cancel PXE boot is for the user to press ESC on the keyboard. Retry will not be attempted due to hardware conditions such as an invalid EEPROM checksum or failing to establish link. Default value is 0.

Bit	Description
10:8	Selects the agent's boot order setup mode. This field changes the agent's default behavior in order to make it compatible with systems that do not completely support the BBS and PnP Expansion ROM standards. Valid values and their meanings are: 000b Normal behavior. The agent will attempt to detect BBS and PnP Expansion ROM support as it normally does. (default) 001b Force Legacy mode. The agent will not attempt to detect BBS or PnP Expansion ROM supports in the BIOS and will assume the BIOS is not compliant. The user can change the BIOS boot order in the Setup Menu. 010b Force BBS mode. The agent will assume the BIOS is BBS-compliant, even though it may not be detected as such by the agent's detection code. The user can NOT change the BIOS boot order in the Setup Menu. 011b Force PnP Int18 mode. The agent will assume the BIOS allows boot order setup for PnP Expansion ROMs and will hook interrupt 18h (to inform the BIOS that the agent is a bootable device) in addition to registering as a BBS IPL device. The user can NOT change the BIOS boot order in the Setup Menu. 100b Force PnP Int19 mode. The agent will assume the BIOS allows boot order setup for PnP Expansion ROMs and will hook interrupt 19h (to inform the BIOS that the agent is a bootable device) in addition to registering as a BBS IPL device. The user can NOT change the BIOS boot order in the Setup Menu. 101b - 111b Reserved for future use. If specified, is treated as a value of 000b.
7:6	Reserved - must be 0
5	Reserved
4	Disable Legacy Wakeup Support. If this bit is set to 1, the user is not allowed to change the Legacy OS Wakeup Support menu option. Default value is 0.
3	Disable Boot Selection. If this bit is set to 1, the user is not allowed to change the boot order menu option. Default value is 0.
2	Disable Protocol Select. If set to 1, the user is not allowed to change the boot protocol. Default value is 0.
1	Disable Title Message. If this bit is set to 1, the title message displaying the version of the Boot Agent is suppressed; the Control-S message is also suppressed. This is for OEMs who do not wish the boot agent to display any messages at system boot. Default value is 0.
0	Disable Setup Menu. If this bit is set to 1, the user is not allowed to invoke the setup menu by pressing Control-S. In this case, the EEPROM may only be changed via an external program. Default value is 0.

24.11.8 PXE Boot Agent Version Number (Word 0x32)

Word 32h of the EEPROM is used to store the version of the boot agent that is stored in the flash image. Some older diagnostic tools read this word in order to report the version of the Boot Agent in the flash.

Note: PCH does not support Flash so this word is not valid.

24.11.9 IBA Capabilities (Word 0x33)

Word 33h of the EEPROM is used to enumerate the boot technologies that have been programmed into the flash. This is updated by flash configuration tools and is not updated or read by IBA.

Note: PCH does not support Flash so this word is not valid.

24.11.10 PXE Reserved Words (Words 0x36, 0x3C, 0x3E)

These words are reserved for other PXE usages.

24.11.11 Alternate MAC Address Pointer (Word 0x37)

This word may point to a location in the EEPROM containing additional MAC addresses used by system management functions. If the additional MAC addresses are not supported, the word shall be set to 0xFFFF. The structure of the alternate MAC address block can be found in Table 24-14.

Table 24-14. Alternate MAC Address Block

Word Offset	Description[1]
0x0 ... 0x2	Alternate MAC Address for LAN port assigned to PCI function 1
0x3 ... 0x5	Alternate MAC Address for LAN port assigned to PCI function 2
0x6 ... 0x8	Alternate MAC Address for LAN port assigned to PCI function 3
0x9 ... 0xB	Alternate MAC Address for LAN port assigned to PCI function 4

1. An alternate MAC Address value of 0xFFFF-FFFF-FFFF means that no alternate MAC address is present for the port.

24.11.12 Checksum Word (Offset 0x3F)

The checksum words (Offset 0x3F from start of Common, LAN 1, LAN 2 and LAN 3 sections) are used to ensure that the base EEPROM image is a valid image. The value of this word should be calculated such that after adding all the words (0x00:0x3E), including the checksum word itself, the sum should be 0xBABA. The initial value in the 16-bit summing register should be 0x0000 and the carry bit should be ignored after each addition.

Note: Hardware does not calculate the checksum word during EEPROM write; it must be calculated by software independently and included in the EEPROM write data. Hardware does not compute a checksum over words 0x00:0x3F during EEPROM reads in order to determine validity of the EEPROM image; this field is provided strictly for software verification of EEPROM validity. All hardware configurations based on word 0x00:0x3F content is based on the validity of the *Signature* field of the *EEPROM Sizing* EEPROM word (*Signature* must be 01b).

24.11.13 Image Unique ID (Word 0x42, 0x43)

These words contain a unique 32-bit ID for each image generated by Intel to enable tracking of images and comparison to the original image if testing a customer EEPROM image.

§ §

25.0 GbE Power Management

This section describes how power management is implemented in the GbE Controller. The Controller supports the Advanced Configuration and Power Interface (ACPI) specification as well as Advanced Power Management (APM).

Note: Power management can be disabled via bits in the *Initialization Control Words*, which is loaded from the EEPROM during power-up reset. Even when disabled, the power management register set is still present. Power management support is required by the PCIe* specification.

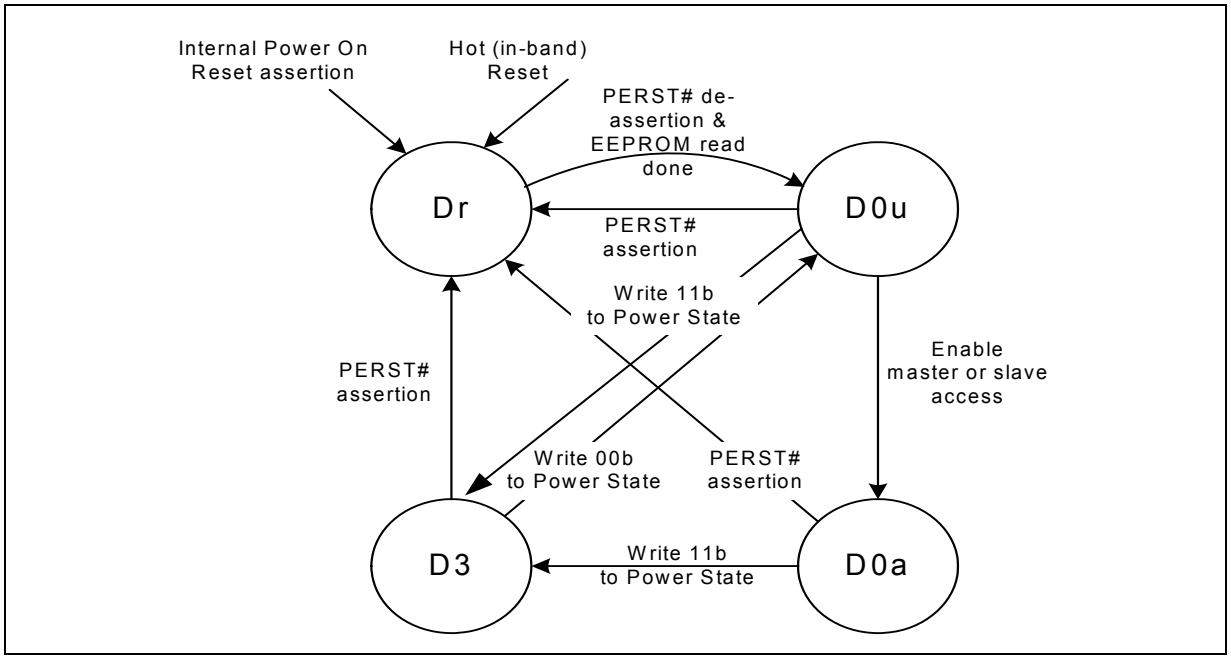
25.1 General Power State Information

25.2 Power States

The Controller supports D0 and D3 (D3 Hot and D3 Cold) power states defined in the PCI Power Management and PCIe* specifications. D0 is divided into two sub-states: D0u (D0 un-initialized), and D0a (D0 active). In addition, the Controller supports a Dr state that is entered when PCIE_EP_RST# is asserted (including the D3cold state).

Figure 25-1 shows the power states and transitions between them.

Figure 25-1. Power Management State Diagram



25.2.1 D0 Uninitialized (D0u) State

The D0u state is a low-power state used after GBE_AUX_PWR_OK (PCIE_EP_RST#) is de-asserted following power-up (cold or warm), on hot reset (in-band reset through PCIe* physical layer message), or on D3 exit.

When entering D0u, the Controller disables wake ups and asserts a reset to the PHY while the EEPROM is being read. If the *APM Mode* bit in the EEPROM's *Initialization Control Word 2* is set, then APM wake up is enabled.

25.2.1.1 Entry Into D0u State

D0u is reached from either the Dr state (on de-assertion of PCIE_EP_RST#) or the D3hot state (by configuration software writing a value of 00b to the *Power State* field of the PCI PM registers).

De-asserting PCIE_EP_RST# means that the entire state of the Controller is cleared, other than sticky bits. State is loaded from the EEPROM, followed by establishment of the PCIe* link. Once this is done, configuration software can access the Controller.

On a transition from D3 to D0u state, the Controller PCI configuration space is not reset. However, the Controller requires that software perform a full re-initialization of the function including its PCI configuration space.

25.2.2 D0 Active (D0a) State

Once memory space is enabled, the Controller enters an active state. It can transmit and receive packets if properly configured by the software device driver. The PHY is enabled or re-enabled by the software device driver to operate/auto-negotiate to full line speed/power if not already operating at full capability. Any APM wake up previously active remains active. The software device driver can deactivate APM wake up by writing to the Wake Up Control (WUC) register or activate other wake-up filters by writing to the Wake Up Filter Control (WUFC) register.

25.2.2.1 Entry to D0a State

D0a is entered from the D0u state by writing a 1b to the *Memory Access Enable* or the *I/O Access Enable* bit of the PCI Command register. The DMA and MAC of the appropriate LAN function are also enabled.

25.2.3 D3 State (PCI-PM D3hot)

The Controller transitions to D3 when the system writes a 11b to the Power State field of the Power Management Control/Status Register (PMCSR). Any wake-up filter settings that were enabled before entering this reset state are maintained. Upon completion or during the transition to D3 state, the Controller clears the *Memory Access Enable* and *I/O Access Enable* bits of the PCI Command register, which disables memory access decode. In D3, the Controller only responds to PCI configuration accesses and does not generate master cycles.

Configuration and message requests are the only TLPs accepted by a function in the D3hot state. All other received requests must be handled as unsupported requests, and all received completions can optionally be handled as unexpected completions. If an error caused by a received TLP (such as an unsupported request) is detected while in D3hot, and reporting is enabled, the link must be returned to L0 if it is not already in L0 and an error message must be sent. See section 5.3.1.4.1 in the PCIe* Base Specification

A D3 state is followed by either a D0u state (in preparation for a D0a state) or by a transition to Dr state (PCI-PM D3cold state). To transition back to D0u, the system writes a 00b to the *Power State* field of the Power Management Control/Status Register (PMCSR). Transition to Dr state is through PCIE_EP_RST# assertion.

25.2.3.1 Entry to D3 State

Transition to D3 state is through a configuration write to the *Power State* field of the *PCI-PM* registers.

Prior to transition from D0 to the D3 state, the software device driver disables scheduling of further tasks to the Controller; it masks all interrupts and does not write to the Transmit Descriptor Tail (*TDT*) register or to the Receive Descriptor Tail (*RDT*) register and operates the master disable algorithm as defined in Section 25.2.3.2.

If wake up capability is needed, the software device driver should set up the appropriate wake up registers and then the system writes a 1b to the *PME_En* bit of the Power Management Control / Status Register (*PMCSR*) or to the Auxiliary (AUX) *Power PM Enable* bit of the PCIe* Device Control register prior to the transition to D3.

As a response to being programmed into D3 state, the Controller suspends scheduling new transactions. The PCIe* EP discards all the completions destined for the Controller and sends dummy completions to the Controller for active read requests. The EP also clears Memory Access Enable and I/O Access Enable bits of the PCIe* Command register which disables memory access decode. Any Receive packets that have not been put in system memory are kept and discarded upon D3 exit. Any transmit packet not yet transmitted can be transmitted or dropped depending on their progress made in the Controller.

25.2.3.2 Master Disable Via CTRL Register

System software can disable master accesses on the PCIe* link by either clearing the PCI *Bus Master* bit or by bringing the function into a D3 state. From that time on, the Controller must not issue master accesses for this function. Due to the full-duplex nature of PCIe*, and the pipelined design in the Controller, it might happen that multiple requests from several functions are pending when the master disable request arrives. The protocol described in this section insures that a function does not issue master requests to the PCIe* link after its *Master Enable* bit is cleared (or after entry to D3 state).

Two configuration bits are provided for the handshake between the Controller function and its software device driver:

- *GIO Master Disable* bit in the Device Control (*CTRL*) register - When the *GIO Master Disable* bit is set, the Controller blocks new master requests by this function. The Controller then proceeds to issue any pending requests by this function. This bit is cleared on master reset (LAN_PWR_GOOD, PCIe* reset and software reset) to enable master accesses.
- *GIO Master Enable Status* bit in the Device Status (*STATUS*) register - Cleared by the Controller when the *GIO Master Disable* bit is set and no master requests are pending by the relevant function and is set otherwise. Indicates that no master requests are issued by this function as long as the *GIO Master Disable* bit is set. The following activities must end before the Controller clears the *GIO Master Enable Status* bit:
  - Master requests by the transmit and receive engines (for both data and MSI/MSIx interrupts).
  - All pending completions to the Controller are received.

In the event of a PCIe* Master disable (Configuration *Command register.BME* set to 0) on a certain function or LAN port or if the function is moved into D3 state during a DMA access, the Controller generates an internal reset to the function and stops all port DMA accesses and interrupts related to the function. Following move to normal operating mode software driver should re-initialize the receive and transmit queues of the relevant port.

Notes: The software device driver sets the *GIO Master Disable* bit when notified of a pending master disable (or D3 entry). The Controller then blocks new requests and proceeds to issue any pending requests by this function. The software device driver then polls the *GIO Master Enable Status* bit. Once the bit is cleared, it is guaranteed that no requests are pending from this function. The software device driver might time out if the *GIO Master Enable Status* bit is not cleared within a given time.

The *GIO Master Disable* bit must be cleared to enable a master request to the PCIe* link. This can be done either through reset or by the software device driver.

25.2.4 Dr State (D3cold)

Transition to Dr state is initiated on several occasions:

- On system power up - Dr state begins with the assertion of the internal power detection circuit and ends with de-assertion of PCIE_EP_RST#.
- On transition from a D0a state - During operation the system might assert PCIE_EP_RST# at any time. In an ACPI system, a system transition to the G2/S5 state causes a transition from D0a to Dr state.
- On transition from a D3 state - The system transitions the Controller into the Dr state by asserting PCIe* PCIE_EP_RST#.

Any wake-up filter settings that were enabled before entering this reset state are maintained.

The system might maintain PCIE_EP_RST# asserted for an arbitrary time. The de-assertion (rising edge) of PCIE_EP_RST# causes a transition to D0u state.

While in Dr state, the Controller might enter one of several modes with different levels of functionality and power consumption. The lower-power modes are achieved when the Controller is not required to maintain any functionality (see Section 25.2.4.1).

25.2.4.1 Dr Disable Mode

The Controller enters a Dr disable mode on transition to D3cold state when it does not need to maintain any functionality. The conditions to enter either state are:

- The Controller (all PCI functions) is in Dr state
- APM WOL is inactive for all LAN functions
- Pass-through manageability is disabled
- ACPI PME is disabled for all PCI functions
- The Controller Power Down En EEPROM bit (word 0x1E, bit 15) is set (default hardware value is disabled).

Entering Dr disable mode is usually done by asserting PCIe* PCIE_EP_RST#. It might also be possible to enter Dr disable mode by reading the EEPROM while already in Dr state. The usage model for this later case is on system power up, assuming that manageability and wake up are not required. Once the Controller enters Dr state on

power-up, the EEPROM is read. If the EEPROM contents determine that the conditions to enter Dr disable mode are met, the Controller then enters this mode (assuming that PCIe* PCIE_EP_RST# is still asserted).

Note: Exiting Dr disable mode can be done by de-asserting PCIE_EP_RST#.

25.2.4.2 Entry to Dr State

Dr entry on platform power-up begins with the assertion of AUX_PWR_GOOD. The EEPROM is read and determines the Controller configuration. If the *APM Enable* bit in the EEPROM's *Initialization Control Word 3* is set, then APM wake up is enabled.

Entering Dr state from D0a state can be done by asserting PCIE_EP_RST#. An ACPI transition to the G2/S5 state is reflected in the Controller transition from D0a to Dr state. The transition can be orderly (such as, user selecting the shut down option), in which case the software device driver might have a chance to intervene. Or, it might be an emergency transition (such as power button override), in which case, the software device driver is not notified.

Note: Transition from D3 (hot) state to Dr state is done by asserting PCIE_EP_RST#.

25.2.4.3 Auxiliary Power Usage

If the *D3COLD_WAKEUP_ADVEN* bit in the *Software Defined Pins Control* EEPROM word is set to 1, the Controller uses the AUX_PWR indication that auxiliary power is available to it, and therefore advertises D3cold wake up support.

If D3cold is supported, the *PME_En* and *PME_Status* bits of the *Power Management Control/Status* Register (PMCSR), as well as their shadow bits in the Wake Up Control (WUC) register are reset only by the power up reset (detection of power rising).

The only effect of setting *AUX_PWR* to 1b is advertising D3cold wake-up support and changing the functionality in reset of *PME_En* and *PME_Status*.

25.2.5 Device Power-Down State

The Controller enters a global power-down state if all of the following conditions are met:

- The Controller *Power Down Enable* EEPROM bit (word 0x1E bit 15) was set (default hardware value is disabled).
- The Controller is in Dr state.
- The link connections of all ports (SerDes) are in power down mode.

25.3 Timing of Power-State Transitions

Note: All reference to PHY in this section also apply to SGMII/SERDES interface. The Controller in the PCH does not have an integrated PHY but all the signals mentioned in this section also apply to SERDES/SGMII.

The following sections give detailed timing for the state transitions. In the diagrams the dotted connecting lines represent the Controller requirements, while the solid connecting lines represent the Controller guarantees.

The timing diagrams are not to scale. The clocks edges are shown to indicate running clocks only and are not to be used to indicate the actual number of cycles for any operation.

25.3.1 Power Up (Off to Dup to D0u to D0a)

Figure 25-2. Power Up (Off to Dup to D0u to D0a)

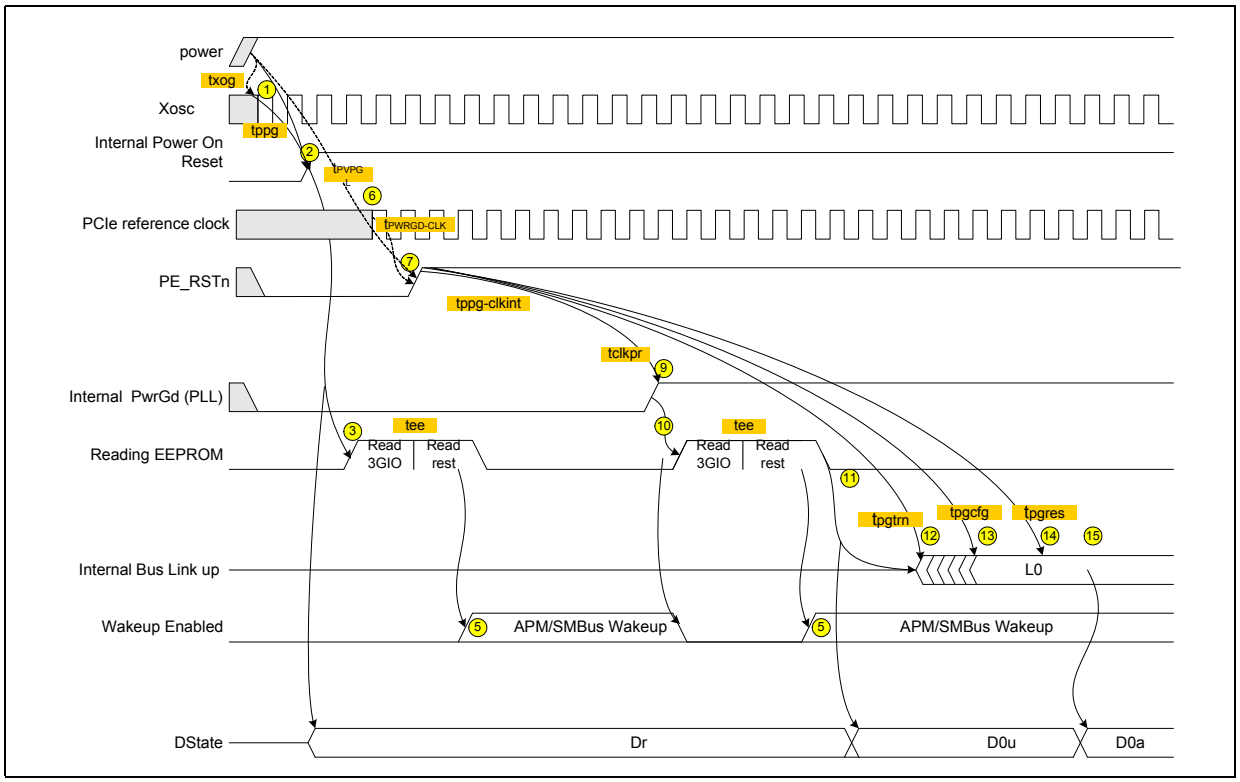


Table 25-1. Power Up (Off to Dup to D0u to D0a)

Note	Description
1	Xosc is stable t_{xog} after power is stable.
2	LAN_PWR_GOOD is asserted after all power supplies are good and t_{ppg} after Xosc is stable.
3	An EEPROM read starts on the rising edge of LAN_PWR_GOOD.
5	APM wake-up mode can be enabled based on what is read from the EEPROM.
7	PCIE_EP_RST# is de-asserted t_{PVPGL} after power is stable (according to PCIe* specification).
9	The PCIe* internal PWRGD signal is asserted t_{clkpr} after the external PCIE_EP_RST# signal.
10	Asserting internal PCIe* PWRGD causes the EEPROM to be re-read, asserts PHY reset, and disables wake up.
11	After reading the EEPROM, PHY reset is de-asserted.
12	Link training starts after t_{pgtrn} from PCIE_EP_RST# de-assertion.
13	A first PCIe* configuration access might arrive after t_{pgcfg} from PCIE_EP_RST# de-assertion.
14	A first PCI configuration response can be sent after t_{pgres} from PCIE_EP_RST# de-assertion.
15	Writing a 1b to the *Memory Access Enable* bit in the PCI Command Register transitions the Controller from D0u to D0. state.

25.3.2 Transition From D0a to D3 and Back Without PCIE_EP_RST#

Figure 25-3. Transition From D0a to D3 and Back Without PCIE_EP_RST#

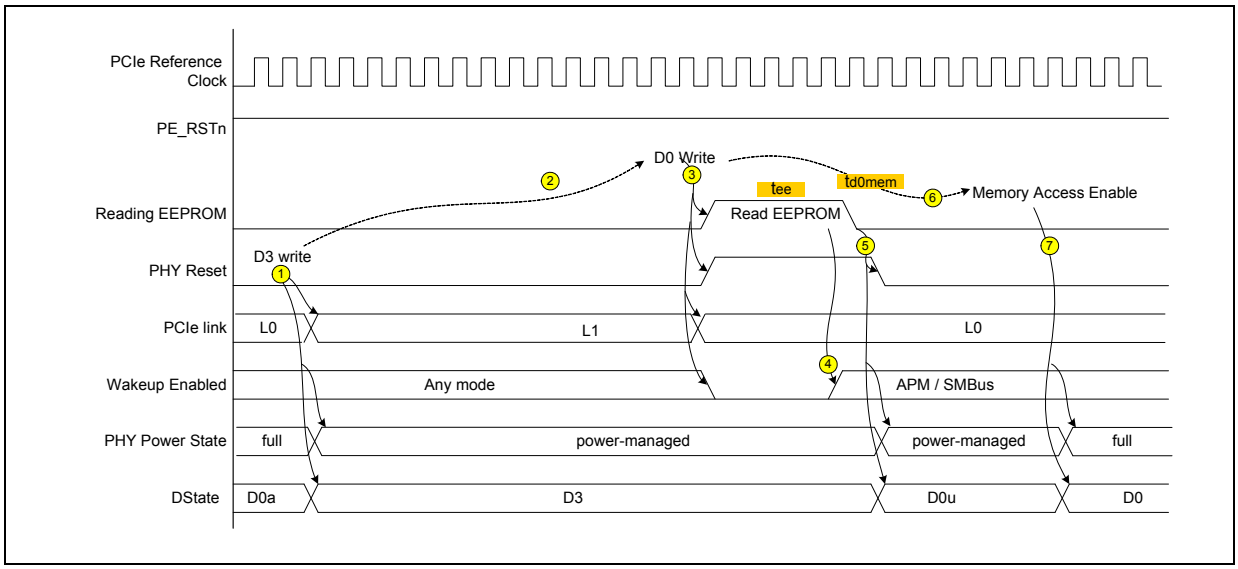


Table 25-2. Transition From D0a to D3 and Back Without PCIE_EP_RST#

Note	Description
1	Writing 11b to the *Power State* field of the Power Management Control/Status Register (PMCSR) transitions the Controller to D3.
2	The system can keep the Controller in D3 state for an arbitrary amount of time.
3	To exit D3 state, the system writes 00b to the *Power State* field of the PMCSR.
4	APM wake-up or SMBus mode might be enabled based on what is read in the EEPROM.
5	After reading the EEPROM, reset to the PHY is de-asserted. The PHY operates at reduced-speed if APM wake up or SMBus is enabled, else powered-down.
6	The system can delay an arbitrary time before enabling memory access.
7	Writing a 1b to the *Memory Access Enable* bit or to the *I/O Access Enable* bit in the PCI Command Register transitions the Controller from D0u to D0 state and returns the PHY to full-power/speed operation.

25.3.3 Transition From D0a to D3 and Back With PCIE_EP_RST#

Figure 25-4. Transition From D0a to D3 and Back With PCIE_EP_RST#

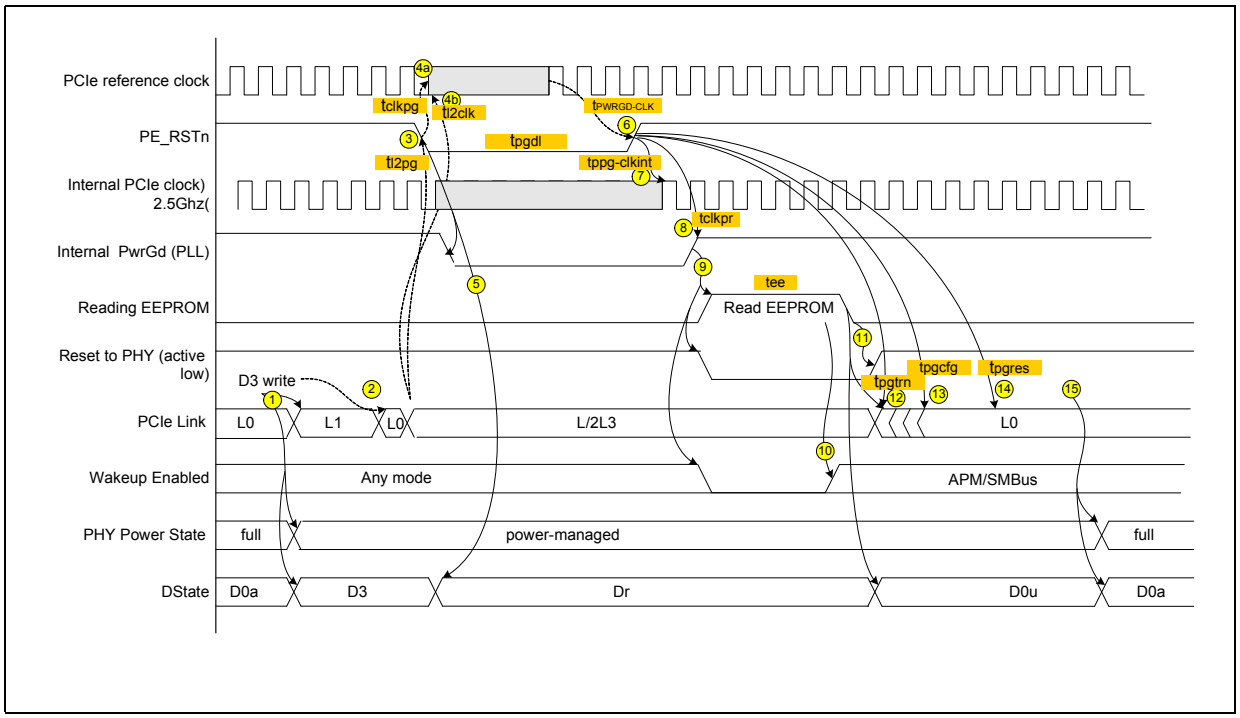


Table 25-3. Transition From D0a to D3 and Back With PCIE_EP_RST#

Note	Description
1	Writing 11b to the *Power State* field of the PMCSR transitions the Controller to D3. PCIe* link transitions to L1 state.
2	The system can delay an arbitrary amount of time between setting D3 mode and transitioning the link to an L2 or L3 state.
3	Following link transition, PCIE_EP_RST# is asserted.
4	The system must assert PCIE_EP_RST# before stopping the PCIe* reference clock. It must also wait t_{l2clk} after link transition to L2/L3 before stopping the reference clock.
5	On assertion of PCIE_EP_RST#, the Controller transitions to Dr state.
6	The system starts the PCIe* reference clock $t_{PE\_RST-CLK}$ before de-assertion PCIE_EP_RST#.
7	The internal PCIe* clock is valid and stable $t_{ppg-clkint}$ from PCIE_EP_RST# de-assertion.
8	The PCIe* internal PWRGD signal is asserted t_{clkpr} after the external PCIE_EP_RST# signal.
9	Asserting internal PCIe* PWRGD causes the EEPROM to be re-read, asserts PHY reset, and disables wake up.
10	APM wake-up mode might be enabled based on what is read from the EEPROM.
11	After reading the EEPROM, PHY reset is de-asserted.
12	Link training starts after t_{pgtrn} from PCIE_EP_RST# de-assertion.

Table 25-3. Transition From D0a to D3 and Back With PCIE_EP_RST# (Continued)

Note	Description
13	A first PCIe* configuration access might arrive after t_{pgcfg} from PCIE_EP_RST# de-assertion.
14	A first PCI configuration response can be sent after t_{pgres} from PCIE_EP_RST# de-assertion.
15	Writing a 1b to the *Memory Access Enable* bit in the PCI Command Register transitions the Controller from D0u to D0 state.

25.3.4 Transition From D0a to Dr and Back Without Transition to D3

Figure 25-5. Transition From D0a to Dr and Back Without Transition to D3

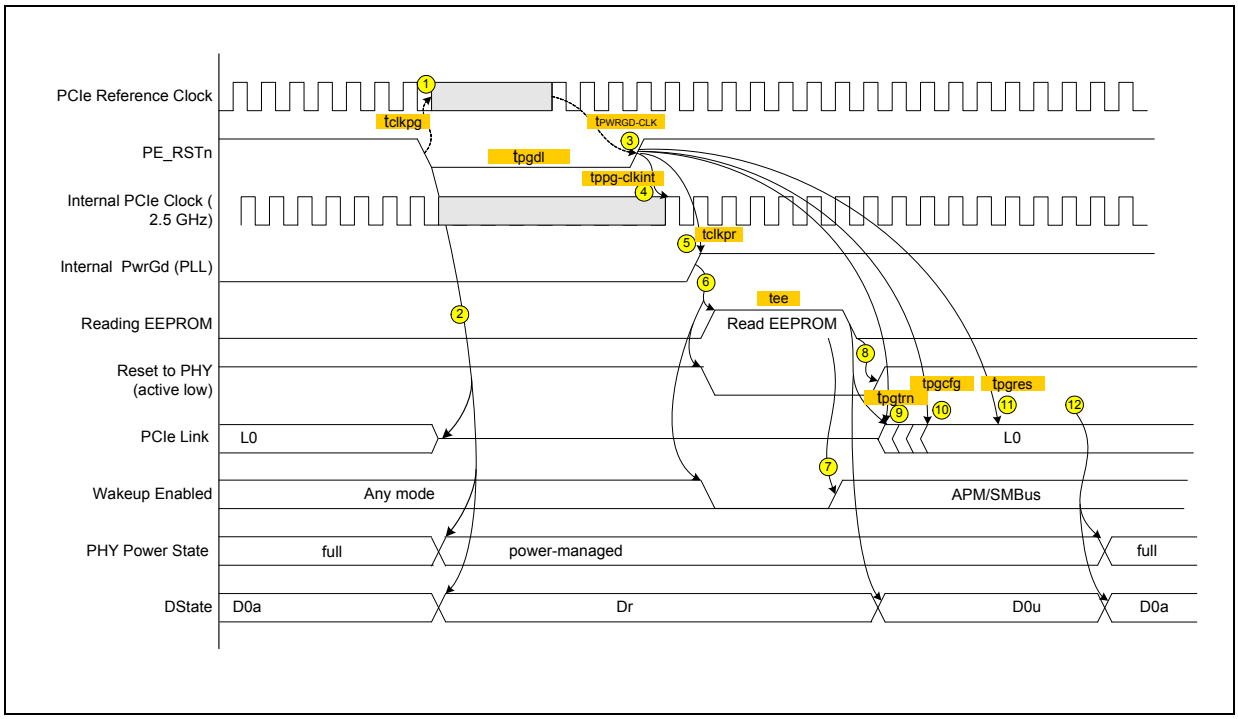


Table 25-4. Transition From D0a to Dr and Back Without Transition to D3

Note	Description
1	The system must assert PCIE_EP_RST# before stopping the PCIe* reference clock. It must also wait t_{l2clk} after link transition to L2/L3 before stopping the reference clock.
2	On assertion of PCIE_EP_RST#, the Controller transitions to Dr state and the PCIe* link transition to electrical idle.
3	The system starts the PCIe* reference clock $t_{PE\_RST-CLK}$ before de-assertion PCIE_EP_RST#.
4	The internal PCIe* clock is valid and stable $t_{ppg-clkint}$ from PCIE_EP_RST# de-assertion.
5	The PCIe* internal PWRGD signal is asserted t_{clkpr} after the external PCIE_EP_RST# signal.
6	Asserting internal PCIe* PWRGD causes the EEPROM to be re-read, asserts PHY reset, and disables wake up.
7	APM wake-up mode might be enabled based on what is read from the EEPROM.

Table 25-4. Transition From D0a to Dr and Back Without Transition to D3 (Continued)

Note	Description
8	After reading the EEPROM, PHY reset is de-asserted.
9	Link training starts after t_{pgtrn} from PCIE_EP_RST# de-assertion.
10	A first PCIe* configuration access might arrive after t_{pgcfg} from PCIE_EP_RST# de-assertion.
11	A first PCI configuration response can be sent after t_{pgres} from PCIE_EP_RST# de-assertion.
12	Writing a 1b to the *Memory Access Enable* bit in the PCI Command Register transitions the Controller from D0u to D0 state.

25.4 Wake Up

The Controller supports two modes of wake-up management:

1. Advanced Power Management (APM) wake up
2. ACPI/PCIe* defined wake up

The usual model is to activate one mode at a time but not both modes together. If both modes are activated, the Controller might wake up the system in unexpected events. For example, if APM is enabled together with PCIe* PME, a magic packet might wake up the system even if APMPME is disabled. Alternatively, if APM is enabled together with some PCIe* filters, packets matching these filters might wake up the system even if PCIe* PME is disabled.

25.4.1 Advanced Power Management Wake Up

Advanced Power Management Wake Up or APM Wakeup (also known as Wake on LAN) is a feature that existed in earlier 10/100 Mb/s NICs. This functionality was designed to receive a broadcast or unicast packet with an explicit data pattern, and then assert a subsequent signal to wake up the system. This was accomplished by using a special signal that ran across a cable to a defined connector on the motherboard. The NIC would assert the signal for approximately 50 ms to signal a wake up. The Controller now uses (if configured) an in-band PM_PME message for this functionality.

On power up, the Controller reads the *APM Enable* bits from the EEPROM *Initialization Control Word 3* into the *APM Enable (APME)* bits of the *Wakeup Control (WUC)* register. These bits control enabling of APM wake up.

When APM wake up is enabled, the Controller checks all incoming packets for Magic Packets. See Section 25.4.3.1.4 for a definition of Magic Packets.

Once the Controller receives a matching magic packet, and if the *Assert PME On APM Wakeup* (*WUC.APMPME*) bit is set in the Wake Up Control (*WUC*) register, it:

- Sets the *PME_Status* bit in the *PMCSR* register and issues a PM_PME message (in most cases, this might require asserting the PE_WAKE_N signal first to resume power and clock to the PCIe* interface).
- Stores the first 128 bytes of the packet in the Wake Up Packet Memory (*WUPM*) register.
- Sets the *Magic Packet Received* bit in the Wake Up Status (*WUS*) register.
- Sets the packet length in the Wake Up Packet Length (*WUPL*) register.

The Controller maintains the first Magic Packet received in the Wake Up Packet Memory (*WUPM*) register until the software device driver writes a 1b to the *Magic Packet Received MAG* bit in the Wake Up Status (*WUS*) register.

APM wake up is supported in all power states and only disabled if a subsequent EEPROM read results in the *APM Wake Up* bit being cleared or software explicitly writes a 0b to the *APM Wake Up* (APME) bit of the *WUC* register.

Note: When *WUC.APMPME* is set PE_WAKE_N is asserted and a PM_PME message is issued even if *PMCSR.PME_En* is cleared. To enable disabling of system Wake-up when *PMCSR.PME_En* is cleared, Software driver should clear the *WUC.APMPME* bit after power-up or PCIe* reset.

25.4.2 PCIe* Power Management Wake Up

The Controller supports PCIe* power management based wake ups. It can generate system wake-up events from three sources:

- Reception of a Magic Packet.
- Reception of a network wakeup packet.
- Detection of a link change of state.

Activating PCIe* power management wake up requires the following:

- The software device driver programs the Wake Up Filter Control (*WUFC*) register to indicate the packets it needs to wake up and supplies the necessary data to the IPv4/v6 Address Table (*IP4AT, IP6AT*) and the Flexible Host Filter Table (*FHFT*). It can also set the *Link Status Change Wake Up Enable* (*LNKC*) bit in the Wake Up Filter Control (*WUFC*) register to cause wake up when the link changes state.
- The operating system (at configuration time) writes a 1b to the *PME_En* bit of the Power Management Control/Status (*PMCSR.8*) register.

Normally, after enabling wake up, the operating system writes 11b to the lower two bits of the *PMCSR* register to place the Controller into low-power mode.

Once wake up is enabled, the Controller monitors incoming packets, first filtering them according to its standard address filtering method, then filtering them with all of the enabled wakeup filters. If a packet passes both the standard address filtering and at least one of the enabled wakeup filters, the Controller:

- Sets the *PME_Status* bit in the *PMCSR*.
- Asserts PE_WAKE_N (if the *PME_En* bit in the PMCSR is set).
- Stores the first 128 bytes of the packet in the *Wakeup Packet Memory (WUPM)* register.
- Sets one or more of the received bits in the *Wake Up Status (WUS)* register. The Controller sets more than one bit if a packet matches more than one filter.
- Sets the packet length in the *Wake Up Packet Length (WUPL)* register.

If enabled, a link state change wake up causes similar results, setting *PME_Status*, asserting *PE_WAKE_N* and setting the *Link Status Changed* (LNKC) bit in the Wake Up Status (WUS) register when the link goes up or down.

The Controller supports the following change described in the PCIe* Base Specification, Rev. 1.1RD (section 5.3.3.4) - On receiving a PME_Turn_Off message, the Controller must block the transmission of PM_PME messages and transmit a PME_TO_Ack message upstream. The Controller is permitted to send a PM_PME message after the Link is returned to an L0 state through LDn.

PE_WAKE_N remains asserted until the operating system either writes a 1b to the *PME_Status* bit of the PMCSR register or writes a 0b to the *PME_En* bit.

After receiving a wake-up packet, the Controller ignores any subsequent wake-up packets until the software device driver clears all of the received bits in the Wake Up Status (WUS) register. It also ignores link change events until the software device driver clears the *Link Status Changed (LNKC)* bit in the *Wake Up Status (WUS)* register.

Note: A wake on link change is not supported when configured to SerDes or 1000BASE-KX mode.

25.4.3 Wake-Up Packets

The Controller supports various wake-up packets using two types of filters:

- Pre-defined filters
- Flexible filters

Each of these filters are enabled if the corresponding bit in the Wake Up Filter Control (*WUFC*) register is set to 1b.

25.4.3.1 Pre-Defined Filters

The following packets are supported by the Controller's pre-defined filters:

- Directed packet (including exact, multicast indexed, and broadcast)
- Magic Packet
- ARP/IPv4 request packet
- Directed IPv4 packet
- Directed IPv6 packet

Each of these filters are enabled if the corresponding bit in the *Wakeup Filter Control (WUFC)* register is set to 1b.

The explanation of each filter includes a table showing which bytes at which offsets are compared to determine if the packet passes the filter.

Note: Both VLAN frames and LLC/SNAP can increase the given offsets if they are present.

25.4.3.1.1 Directed Exact Packet

The Controller generates a wake-up event after receiving any packet whose destination address matches one of the 24 valid programmed receive addresses, if the *Directed Exact Wake Up Enable* bit is set in the Wake Up Filter Control (*WUFC.EX*) register.

25.4.3.1.2 Directed Multicast Packet

For multicast packets, the upper bits of the incoming packet's destination address index a bit vector, the Multicast Table Array (*MTA*) that indicates whether to accept the packet. If the *Directed Multicast Wake Up Enable* bit set in the Wake Up Filter Control (*WUFC.MC*) register and the indexed bit in the vector is one, then the Controller generates a wake-up event. The exact bits used in the comparison are programmed by software in the *Multicast Offset* field of the Receive Control (*RCTL.MO*) register.

25.4.3.1.3 Broadcast

If the *Broadcast Wake Up Enable* bit in the Wake Up Filter Control (WUFC.BC) register is set, the Controller generates a wake-up event when it receives a broadcast packet.

25.4.3.1.4 Magic Packet

Magic packets are defined in:

http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/20213.pdf as:

"Once the LAN controller has been put into the Magic Packet mode, it scans all incoming frames addressed to the node for a specific data sequence.This sequence indicates to the controller that this is a Magic Packet frame. A Magic Packet frame must also meet the basic requirements for the LAN technology chosen, such as SOURCE ADDRESS, DESTINATION ADDRESS (which may be the receiving station's IEEE address or a MULTICAST address which includes the BROADCAST address), and CRC. The specific data sequence consists of 16 repetitions of the IEEE address of this node, with no breaks or interruptions. This sequence can be located anywhere within the packet, but must be preceded by a synchronization stream. The synchronization stream allows the scanning state machine to be much simpler. The synchronization stream is defined as 6 bytes of 0xFF. The device will also accept a BROADCAST frame, as long as the 16 repetitions of the IEEE address match the address of the machine to be awakened."

The Controller expects the destination address to either:

- Be the broadcast address (FF.FF.FF.FF.FF.FF)
- Match the value in Receive Address 0 (*RAH0, RAL0*) register. This is initially loaded from the EEPROM but can be changed by the software device driver.
- Match any other address filtering (*RAH*[n], *RAL*[n]) enabled by the software device driver.

The Controller searches for the contents of Receive Address 0 (*RAH0, RAL0*) register as the embedded IEEE address. It considers any non-0xFF byte after a series of at least 6 0xFFs to be the start of the IEEE address for comparison purposes. For example, it catches the case of 7 0xFFs followed by the IEEE address). As soon as one of the first 96 bytes after a string of 0xFFs don't match, it continues to search for another set of at least 6 0xFFs followed by the 16 copies of the IEEE address later in the packet. This definition precludes the first byte of the destination address from being FF.

A Magic Packet's destination address must match the address filtering enabled in the configuration registers with the exception that broadcast packets are considered to match even if the *Broadcast Accept* bit of the *Receive Control (RCTL.BAM)* register is 0b. If APM wake up (wake up by a Magic Packet) is enabled in the EEPROM, the Controller starts up with the Receive Address 0 (*RAH0, RAL0*) register loaded from the EEPROM. This enables the Controller to accept packets with the matching IEEE address before the software device driver loads.

Table 25-5. Magic Packet Structure

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	4	Type		Skip	
Any	6	Synchronizing Stream	FF*6+	Compare	
any+6	96	16 copies of Node Address	A*16	Compare	Compared to Receive Address 0 (RAH0, RAL0) register.

25.4.3.1.5 ARP/IPv4 Request Packet

The Controller supports receiving ARP request packets for wake up if the *ARP* bit is set in the Wake Up Filter Control (*WUFC*) register. Four IPv4 addresses are supported, which are programmed in the IPv4 Address Table (*IP4AT*). A successfully matched packet must contain a broadcast MAC address, a protocol type of 0x0806, an ARP op-code of 0x01, and one of the four programmed IPv4 addresses. The Controller also handles ARP request packets that have VLAN tagging on both Ethernet II and Ethernet SNAP types.

Table 25-6. ARP Packet Structure and Processing

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	2	Type	0x0806	Compare	ARP
14	2	HW Type	0x0001	Compare	
16	2	Protocol Type	0x0800	Compare	
18	1	Hardware Size	0x06	Compare	
19	1	Protocol Address Length	0x04	Compare	
20	2	Operation	0x0001	Compare	
22	6	Sender HW Address	-	Ignore	
28	4	Sender IP Address	-	Ignore	
32	6	Target HW Address	-	Ignore	
38	4	Target IP Address	IP4AT	Compare	May match any of four values in IP4AT.

25.4.3.1.6 Directed Ipv4 Packet

The Controller supports receiving directed IPv4 packets for wake up if the *IPV4* bit is set in the Wake Up Filter Control (*WUFC*) register. Four IPv4 addresses are supported, which are programmed in the IPv4 Address Table (*IP4AT*). A successfully matched packet must contain the station's MAC address, a protocol type of 0x0800, and one of the four programmed IPv4 addresses. The Controller also handles directed IPv4 packets that have VLAN tagging on both Ethernet II and Ethernet SNAP types.

Table 25-7. IPv4 Packet Structure and Processing

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	2	Type	0x0800	Compare	IP
14	1	Version/ HDR length	0x4X	Compare	Check IPv4
15	1	Type of Service	-	Ignore	
16	2	Packet Length	-	Ignore	

Table 25-7. IPv4 Packet Structure and Processing (Continued)

18	2	Identification	-	Ignore	
20	2	Fragment Info	-	Ignore	
22	1	Time to live	-	Ignore	
23	1	Protocol	-	Ignore	
24	2	Header Checksum	-	Ignore	
26	4	Source IP Address	-	Ignore	
30	4	Destination IP Address	IP4AT	Compare	May match any of four values in IP4AT.

25.4.3.1.7 Directed IPv6 Packet

The Controller supports receiving directed IPv6 packets for wake up if the *IPV6* bit is set in the *Wake Up Filter Control (WUFC)* register. One IPv6 address is supported and is programmed in the *IPv6 Address Table (IP6AT)*. A successfully matched packet must contain the station's MAC address, a protocol type of 0x86DD, and the programmed IPv6 address. In addition, the *IPAV.V60* bit should be set. The Controller also handles directed IPv6 packets that have VLAN tagging on both Ethernet II and Ethernet SNAP types.

Table 25-8. IPv6 Packet Structure and Processing

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	2	Type	0x86DD	Compare	IP
14	1	Version/ Priority	0x6X	Compare	Check IPv6
15	3	Flow Label	-	Ignore	
18	2	Payload Length	-	Ignore	
20	1	Next Header	-	Ignore	
21	1	Hop Limit	-	Ignore	
22	16	Source IP Address	-	Ignore	
38	16	Destination IP Address	IP6AT	Compare	Match value in IP6AT.

25.4.3.2 Flexible Filters

The Controller supports a total of 8 flexible filters. Each filter can be configured to recognize any arbitrary pattern within the first 128 bytes of the packet. To configure the flexible filters, software programs the mask values (required values and the minimum packet length), into the Flexible Host Filter Table (FHFT and FHFT_EXT). These 8 flexible filters contain separate values for each filter. Software must also enable the filters in the Wake Up Filter Control (*WUFC*) register, and enable the overall wake up functionality. The overall wake up functionality must be enabled by setting *PME_En* in the PMCSR or the Wake Up Control (WUC) register.

Once enabled, the flexible filters scan incoming packets for a match. If the filter encounters any byte in the packet where the mask bit is one and the byte doesn't match the value programmed in the Flexible Host Filter Table (*FHFT* or *FHFT_EXT*), then

the filter fails that packet. If the filter reaches the required length without failing the packet, it passes the packet and generates a wake-up event. It ignores any mask bits set to one beyond the required length.

Note: The flex filters are temporarily disabled when read from or written to by the host. Any packet received during a read or write operation is dropped. Filter operation resumes once the read or write access completes.

The following packets are listed for reference purposes only. The flexible filter could be used to filter these packets.

25.4.3.2.1 IPX Diagnostic Responder Request Packet

An IPX diagnostic responder request packet must contain a valid MAC address, a protocol type of 0x8137, and an IPX diagnostic socket of 0x0456. It might include LLC/SNAP headers and VLAN tags. Since filtering this packet relies on the flexible filters, which use offsets specified by the operating system directly, the operating system must account for the extra offset LLC/SNAP headers and VLAN tags.

Table 25-9. IPX Diagnostic Responder Request Packet Structure and Processing

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	2	Type	0x8137	Compare	IPX
14	16	Some IPX Stuff	-	Ignore	
30	2	IPX Diagnostic Socket	0x0456	Compare	

25.4.3.2.2 Directed IPX Packet

A valid directed IPX packet contains the station's MAC address, a protocol type of 0x8137, and an IPX node address that is equal to the station's MAC address. It might include LLC/SNAP headers and VLAN tags. Since filtering this packet relies on the flexible filters, which use offsets specified by the operating system directly, the operating system must account for the extra offset LLC/SNAP headers and VLAN tags.

Table 25-10. IPX Packet Structure and Processing

Offset	# of bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	8	Possible LLC/SNAP Header		Skip	
12	4	Possible VLAN Tag		Skip	
12	2	Type	0x8137	Compare	IPX
14	10	Some IPX Info	-	Ignore	
24	6	IPX Node Address	Receive Address 0	Compare	Must match receive address 0.

25.4.3.2.3 IPv6 Neighbor Discovery Filter

In IPv6, a neighbor discovery packet is used for address resolution. A flexible filter can be used to check for a neighborhood discovery packet.

25.4.3.2.4 Utilizing Flex Wake-Up Filters In Normal Operation

The Controller enables utilizing the WoL Flex filters in normal operation, when in D0 power management state, for queuing decisions. Further information is in Section 26.1.1.6.

25.4.3.3 Wake Up Packet Storage

The Controller saves the first 128 bytes of the wake-up packet in its internal buffer, which can be read through the *Wake Up Packet Memory (WUPM)* register after the system wakes up.

§ §

26.0 GbE Inline Functions

26.1 Receive Functionality

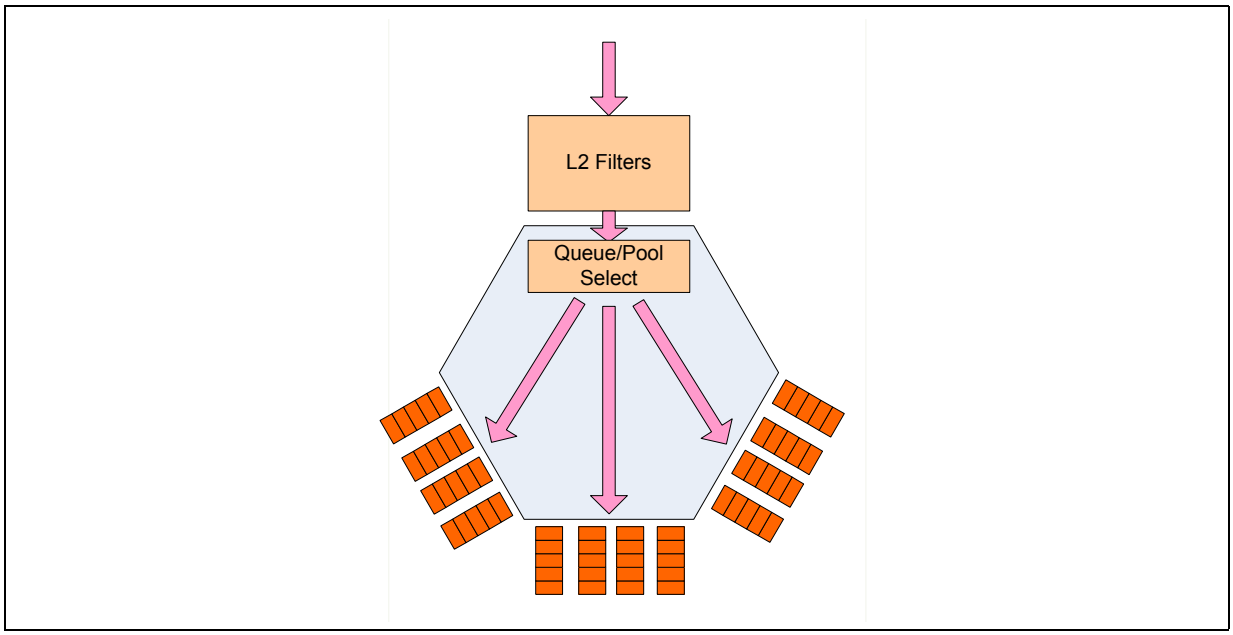
26.1.1 Receive Queues Assignment

A received packet goes through three stages of filtering as shown in Figure 26-1. Figure 26-1 describes a switch-like structure that is used in virtualization mode to route packets between the network port (top of drawing) and one or more virtual ports (bottom of figure), where each virtual port can be associated with a virtual machine, a VMM or any other software entity.

The first step in queue assignment is to verify that the packet is destined to the port. This is done by a set of L2 filters as described in Section 26.1.2.

The second stage is specific to virtualization environments and defines the virtual ports (called pools) that are the targets for the Rx packet. A packet can be associated with any number of ports/pools using a selection process described in Section 26.1.1.2.

Figure 26-1. Stages in Packet Filtering



In the third stage, a received packet that successfully passed the Rx filters is associated with one or more receive descriptor queues as described in this section.

The following filter mechanisms determines the destination of a receive packet. These are described briefly in this section and in full details in separate sections:

- Virtualization — In a virtualized environment, DMA resources are shared between more than one software entity (operating system and/or software device driver). This is done by allocating receive descriptor queues/pools to virtual partitions (VMM or VMs). Virtualization assigns to each received packet one or more pool indices. Packets are routed to a pool based on their pool index and other considerations. See Section 26.1.1.2 for details on routing for virtualization.
- RSS — Receive Side Scaling distributes packet processing between several processor cores by assigning packets into different descriptor queues. RSS assigns to each received packet an RSS index. Packets are routed to a queue out of a set of Rx queues based on their RSS index and other considerations. See Section 26.1.1.8 for details on RSS.
- L2 Ethertype filters — These filters identify packets by their L2 Ether type and assign them to receive queues. Examples of possible uses are LLDP packets and 802.1X packets. See Section 26.1.1.4 for mode details. The Controller incorporates 8 Ether-type filters per port.
- filters — These filters identify . Each filter consists of a (protocol and destination TCP/UDP port) and routes packets into one of the Rx queues. The Controller has 8 such filters per port. See Section 26.1.1.5 for details.
- TCP SYN filters — The Controller might route TCP packets with their *SYN* flag set into a separate queue. *SYN* packets are often used in *SYN* attacks to load the system with numerous requests for new connections. By filtering such packets to a separate queue, security software can monitor and act on *SYN* attacks. The Controller has one such filter per port. See Section 26.1.1.7 for more details.
- Flex Filters — These filters can be either used as WoL filters when the Controller is in D3 state or for queueing in normal operating mode (D0 state). Filters enable queueing according to a match of any 128 Byte sequence at the beginning of a packet. Each one of the 128 bytes can be either compared or masked using a dedicated mask field. The Controller has 8 such filters per port. See Section 26.1.1.6 for details.

Typically, packet reception consists of recognizing the presence of a packet on the wire, performing address filtering, storing the packet in the receive data FIFO, transferring the data to one of the 16 receive queues in host memory, and updating the state of a receive descriptor.

Note: Maximum supported received-packet size is 9.5 KB (9728 bytes).

A received packet is allocated to a queue based on the previous criteria and the following order:

1. Queue by L2 Ether-type filters (if a match)
2. If RFCTL.SYNQFP is 0b (filter and Flex filter have priority), then:
   a. Queue by Flex filter (if a match)
   b. Queue by filter
   c. Queue by SYN filter (if a match)
3. If RFCTL.SYNQFP is 1b (SYN filter has priority), then:
   a. Queue by SYN filter (if a match)
   b. Queue by Flex filter (if a match)
   c. Queue by filter (if a match)
4. Define a pool (if virtualization enabled)
5. Queue by RSS (if RSS enabled).

26.1.1.1 Queuing in a Non-Virtualized Environment

When the *MRQC.Multiple Receive Queues Enable* field equals 010b (Multiple receive queues as defined by filters and RSS for 8 queues) the received packet is assigned to a queue in the following manner:

- L2 Ether-type filters — Each filter identifies one of 16 receive queues.
- SYN filter — Identifies one of 16 receive queues.
- Flex Filter - Each filter identifies one of 16 receive queues.
- filters — Each filter identifies one of 16 receive queues.
- RSS filters - Identifies one of 2 x 16 queues through the RSS index. The following modes are supported:
  - No RSS — The default queue as defined in *MRQC.DEF_Q* is used for packets that do not meet any of the previous conditions.
  - RSS only — A set of 16 queues is allocated for RSS. The queue is identified through the RSS index. It is possible to use a subset of the 16 queues.

When the *MRQC.Multiple Receive Queues Enable* field equals 000b (Multiple receive queues as defined by filters) the received packet is assigned to a queue in the following manner:

- L2 Ether-type filters — Each filter identifies one of 16 receive queues.
- SYN filter — Identifies one of 16 receive queues.
- Flex Filter - Each filter identifies one of 16 receive queues.
- filters — Each filter identifies one of 16 receive queues.

26.1.1.2 Receive Queuing in a Virtualized Environment

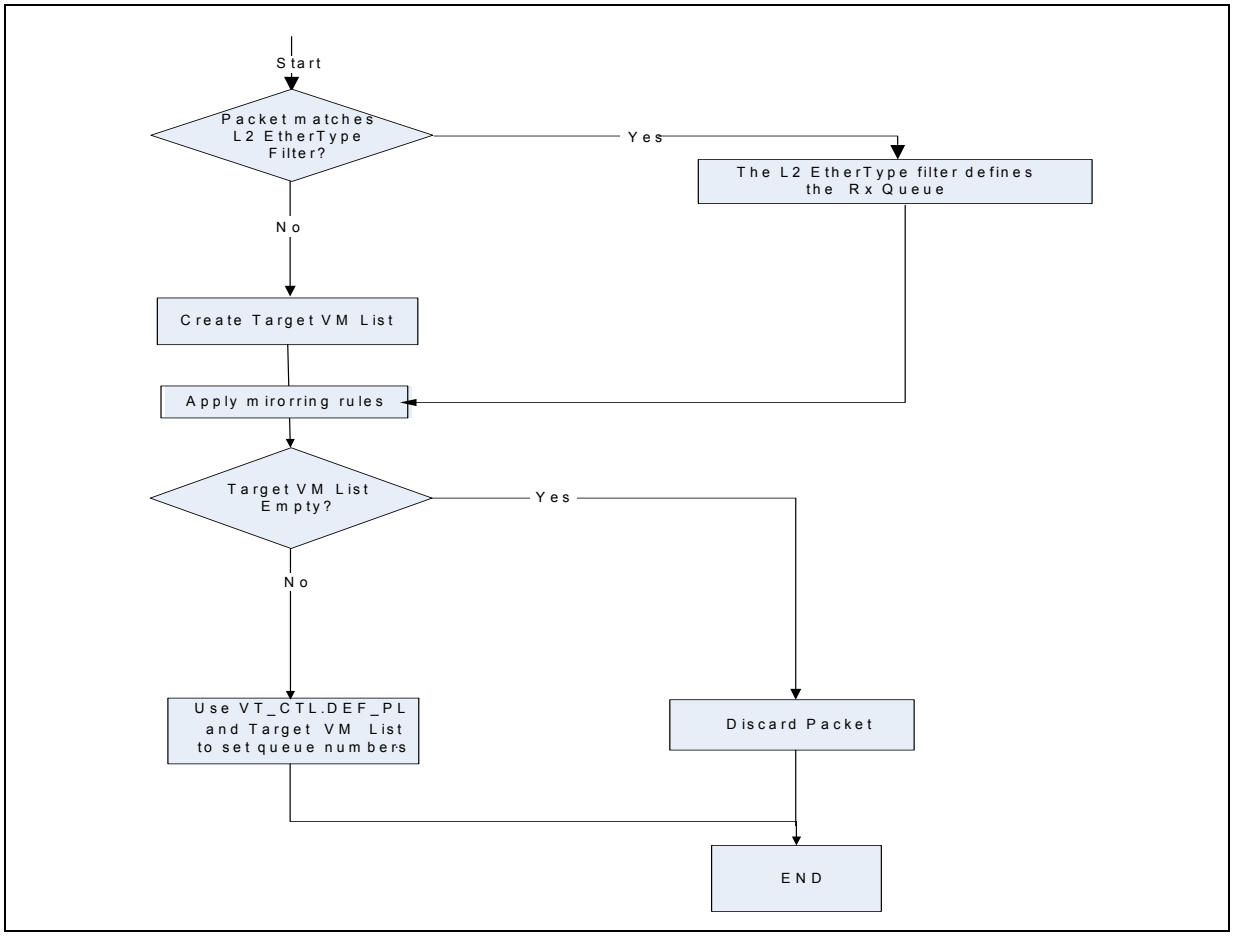
In VMDq mode, system software allocates the pools to the VMM, an IOVM, or to VMs. When the *MRQC.Multiple Receive Queues Enable* field equals 011b (Multiple receive queues as defined by VMDq), the received packets are allocated to the 16 receive queues/pools in the following manner:

Incoming packets are associated with pools/ queues based on their L2 characteristics as described in Section 26.8.3. This section describes the following stage, where an Rx queue is assigned to each replication of the Rx packet as determined by its pool's association.

A received packet is assigned to a queue/pool in the following manner:

- L2 Ether-type filters — Each filter identifies a specific receive queue (the queue is usually allocated to the VMM or a service operating system).
- SYN filter — Not supported in VT modes.
- Not supported Flex filters —Not supported in VT mode.
- RSS filters - Not supported in VT mode

Figure 26-2. Receive Queuing Flow (Virtualization)



26.1.1.3 Queue Configuration Registers

Configuration registers (CSRs) that control queue operation are replicated per queue (total of 16 copies of each register per port). Each of the replicated registers correspond to a queue such that the queue index equals the serial number of the register (such as register 0 corresponds to queue 0, etc.). Registers included in this category are:

- RDBAL and RDBAH — Rx Descriptor Base
- RDLEN — RX Descriptor Length
- RDH — RX Descriptor Head
- RDT — RX Descriptor Tail
- RXDCTL — Receive Descriptor Control
- RXCTL — Rx DCA Control
- SRRCTL — Split and Replication Receive Control
- PSRTYPE — Packet Split Receive type

26.1.1.4 L2 Ether-Type Filters

These filters identify packets by L2 Ether-type and assign them to a receive queue. The following usages have been identified:

- IEEE 802.1X packets — Extensible Authentication Protocol over LAN (EAPOL).
- Time sync packets (such as IEEE 1588) — Identifies Sync or Delay_Req packets
- IEEE802.1AB LLDP (Link Layer Discovery Protocol) packets.

The Controller incorporates 8 Ether-type filters.

The *Packet Type* field in the Rx descriptor captures the filter number that matched the L2 Ether-type. See Section 26.1.5 for decoding of the *Packet Type* field.

The Ether-type filters are configured via the ETQF register as follows:

- The *EType* field contains the 16-bit Ether-type compared against all L2 type fields in the Rx packet.
- The *Filter Enable* bit enables identification of Rx packets by Ether-type according to this filter. If this bit is cleared, the filter is ignored for all purposes.
- The *Rx Queue* field contains the absolute destination queue for the packet.
- The *1588 Time Stamp* field indicates that the packet should be time stamped according to the IEEE 1588 specification.
- The *Queue Enable* field enables forwarding Rx packets based on the Ether-type defined in this register.

Note: Software should not assign the same Ether-type value to different ETQF filters with different *Rx Queue* assignments.

Special considerations for Virtualization modes:

- Packets that match an Ether-type filter are diverted from their original pool (the pool identified by the L2 filters) to the pool used as the pool to which the queue in the *Queue* field belongs. In other words, The L2 filters are ignored in determining the pool for such packets.
- The same applies for multi-cast packets. A single copy is posted to the pool defined by the filter.
- Mirroring rules:
  - If a pool is being mirrored, the pool to which the queue belongs to is used to determine if a packet that matches the filter should be mirrored.
  - The queue inside the pool is used for both the original pool and the mirroring pool.

26.1.1.5 Filters

These filters identify specific. Each filter consists of a (protocol and destination TCP/UDP port) and forwards packets into one of the receive queues.

The Controller incorporates 8 such filters.

The filters are configured via the *TTQF* (See Section 28.1.3), *IMIR* (See Section 28.10.1.1) and *IMIR_EXT* (See Section 28.10.1.2) registers as follows (per filter):

- Protocol — Identifies the IP protocol, part of the queue filters. Enabled by a bit in the field.

- Destination port — Identifies the TCP/UDP destination port, part of the queue filters. Enabled by the *IMIR.PORT_BP* bit.
- Size threshold — Identifies the length of the packet that should trigger the filter. This is the length as received by the host, not including any part of the packet removed by hardware. Enabled by the *IMIREXT.Size_BP* field.
- Control Bits — Identify TCP flags that might be part of the filtering process. Enabled by the *IMIREXT.CtrlBit_BP* field.
- Rx queue — Determines the Rx queue for packets that match this filter:
  - The *.Rx Queue* field contains the queue serial number.
- Queue enable — Enables forwarding a packet that uses this filter to the queue defined in the *.Rx Queue* field.
- Mask — A -bit field that masks. The filter is a logical AND of the non-masked fields. If all fields are masked, the filter is not used for queue forwarding.

Notes: If more than one filter with the same priority is matched by the packet, the first filter (lowest ordinal number) is used in order to define the queue destination of this packet.

- The immediate interrupt and 1588 actions are defined by the OR of all the matching filters.

26.1.1.6 Flex Filters

The Controller supports a total of 8 flexible filters. Each filter can be configured to recognize any arbitrary pattern within the first 128 bytes of the packet. To configure the flexible filters, software programs the mask values (required values and the minimum packet length), into the Flexible Host Filter Table (*FHFT* and *FHFT_EXT*, See Section 28.20.2 and Section 28.20.3). These 8 flexible filters can be used as for wake-up when in D3 state or for queueing when in D0 state. Software must enable the filters in the *Wake Up Filter* Control (*WUFC See* Section 28.20.1.2) register for operation in D3 or D0 mode. In D0 mode these filters enable forwarding of packets that match up to 128 Bytes defined in the filter to one of the receive queues. In D3 mode these filters can be used for Wake-on-Lan as described in Section 25.4.3.2.

Once enabled, the flexible filters scan incoming packets for a match. If the filter encounters any byte in the packet where the mask bit is one and the byte doesn't match the value programmed in the Flexible Host Filter Table (*FHFT* or *FHFT_EXT*), then the filter fails that packet. If the filter reaches the required length without failing the packet, it forwards the packet to the appropriate receive queue. It ignores any mask bits set to one beyond the required length (defined in the Length field in the *FHFT* or *FHFT_EXT* registers).

Note: The flex filters are temporarily disabled when read from or written to by the host. Any packet received during a read or write operation is dropped. Filter operation resumes once the read or write access completes.

The flex fitters are configured in D0 state via the *WUFC*, *FHFT* and *FHFT_EXT* registers as follows (per filter):

- Byte Sequence to be compared — Program 128 Byte sequence, mask bits and *Length* field in FHFT and *FHFT_EXT* registers.
- Filter Priority — Program filter priority in queueing field in *FHFT* and *FHFT_EXT* registers.
- Receive queue — Program receive queue to forward packet in queueing field in *FHFT* and *FHFT_EXT* registers.

- Filter actions — Program immediate interrupt requirement in queueing field in *FHFT* and *FHFT_EXT* registers.
- Filter enable — Set *WUFC.FLEX_HQ* bit to 1 to enable flex filter operation in D0 state. Set appropriate *WUFC.FLX[n]* bit to 1 to enable specific flex filter.

Before entering D3 state software device driver programs the *FHFT* and *FHFT_EXT* filters for appropriate wake events and enables relevant filters by setting the *WUFC.FLX[n]* bit to 1. Following move to D0 state the software device driver programs the FHFT and FHFT_EXT filters for appropriate queueing decisions enables relevant filters by setting the *WUFC.FLX[n]* bit to 1 and the *WUFC.FLEX_HQ* bit to 1.

Notes: If more than one flex filter with the same priority is matched by the packet, the first filter (lowest address) is used in order to define the queue destination of this packet.

The immediate interrupt action is defined by the OR of all the matching filters.

26.1.1.7 SYN Packet Filters

The Controller might forward TCP packets whose *SYN* flag is set into a separate queue. SYN packets are often used in SYN attacks to load the system with numerous requests for new connections. By filtering such packets to a separate queue, security software can monitor and act on SYN attacks.

SYN filters are configured via the SYNQF registers as follows:

- Queue En — Enables forwarding of SYN packets to a specific queue.
- Rx Queue field — Contains the destination queue for the packet.

This filter is not to be used in a virtualized environment.

26.1.1.8 Receive-Side Scaling (RSS)

RSS is a mechanism to distribute received packets into several descriptor queues. Software then assigns each queue to a different processor, sharing the load of packet processing among several processors.

The Controller uses RSS as one ingredient in its packet assignment policy (the others are the various filters and virtualization). The RSS output is a RSS index. The Controller's global assignment uses these bits (or only some of the LSB bits) as part of the queue number.

RSS is enabled in the *MRQC* register. The RSS *Status* field in the descriptor write-back is enabled when the *RXCSUM.PCSD* bit is set (fragment checksum is disabled). RSS is therefore mutually exclusive with UDP fragmentation. Also, support for RSS is not provided when legacy receive descriptor format is used.

When RSS is enabled, the Controller provides software with the following information as required by Microsoft* RSS specification or for device driver assistance:

- A Dword result of the Microsoft* RSS hash function, to be used by the stack for flow classification, is written into the receive packet descriptor (required by Microsoft* RSS).
- A 4-bit RSS *Type* field conveys the hash function used for the specific packet (required by Microsoft* RSS).

Figure 26-3 shows the process of computing an RSS output:

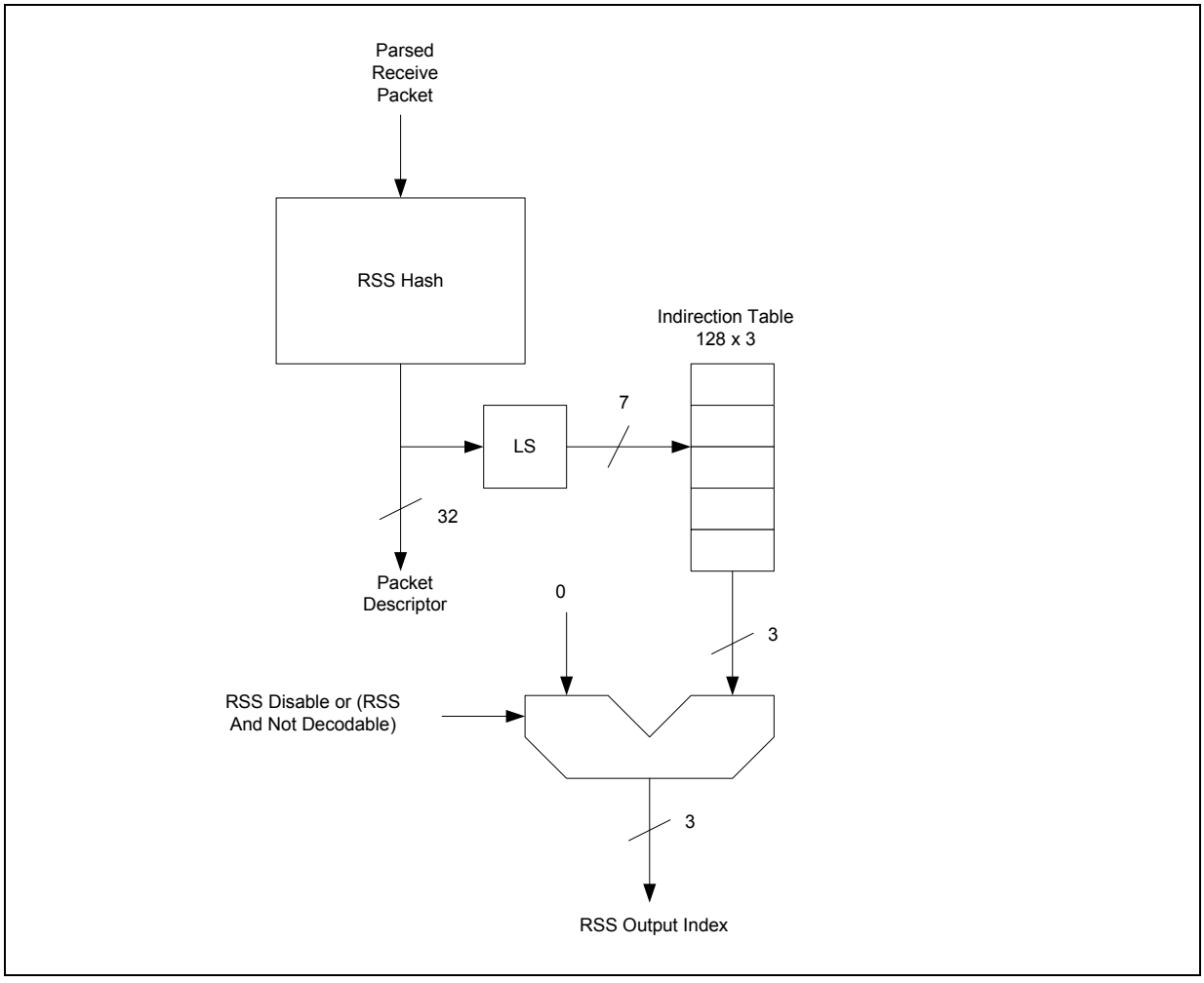
1. The receive packet is parsed into the header fields used by the hash operation (such as IP addresses, TCP port, etc.).

2. A hash calculation is performed. The Controller supports a single hash function, as defined by Microsoft* RSS. The Controller does not indicate to the software device driver which hash function is used. The 32-bit result is fed into the packet receive descriptor.
3. The seven LSB bits of the hash result are used as an index into a 128-entry indirection table. Each entry provides a 3-bit RSS output index.

When RSS is disabled, packets are assigned an RSS output index = zero. System software might enable or disable RSS at any time. While disabled, system software might update the contents of any of the RSS-related registers.

When multiple requests queues are enabled in RSS mode, un-decodable packets are assigned an RSS output index = zero. The 32-bit tag (normally a result of the hash function) equals zero.

Figure 26-3. RSS Block Diagram



26.1.1.8.1 RSS Hash Function

Section 26.1.1.8.1 provides a verification suite used to validate that the hash function is computed according to Microsoft nomenclature.

The Controller hash function follows Microsoft definition. A single hash function is defined with several variations for the following cases:

- TcpIPv4 — The Controller parses the packet to identify an IPv4 packet containing a TCP segment per the criteria described later in this section. If the packet is not an IPv4 packet containing a TCP segment, RSS is not done for the packet.
- IPv4 — The Controller parses the packet to identify an IPv4 packet. If the packet is not an IPv4 packet, RSS is not done for the packet.
- TcpIPv6 — The Controller parses the packet to identify an IPv6 packet containing a TCP segment per the criteria described later in this section. If the packet is not an IPv6 packet containing a TCP segment, RSS is not done for the packet.
- TcpIPv6Ex — The Controller parses the packet to identify an IPv6 packet containing a TCP segment with extensions per the criteria described later in this section. If the packet is not an IPv6 packet containing a TCP segment, RSS is not done for the packet. Extension headers should be parsed for a *Home-Address-Option* field (for source address) or the *Routing-Header-Type-2* field (for destination address).
- IPv6Ex — The Controller parses the packet to identify an IPv6 packet. Extension headers should be parsed for a *Home-Address-Option* field (for source address) or the *Routing-Header-Type-2* field (for destination address). The packet is not required to contain any of these extension headers to be hashed by this function. In this case, the IPv6 hash is used. If the packet is not an IPv6 packet, RSS is not done for the packet.
- IPv6 — The Controller parses the packet to identify an IPv6 packet. If the packet is not an IPv6 packet, receive-side-scaling is not done for the packet.

The following additional cases are not part of the Microsoft RSS specification:

- UdpIPV4 — The Controller parses the packet to identify a packet with UDP over IPv4.
- UdpIPV6 — The Controller parses the packet to identify a packet with UDP over IPv6.
- UdpIPV6Ex — The Controller parses the packet to identify a packet with UDP over IPv6 with extensions.

A packet is identified as containing a TCP segment if all of the following conditions are met:

- The transport layer protocol is TCP (not UDP, ICMP, IGMP, etc.).
- The TCP segment can be parsed (such as IP options can be parsed, packet not encrypted).
- The packet is not fragmented (even if the fragment contains a complete TCP header).

Bits[31:16] of the Multiple Receive Queues Command (*MRQC)* register enable each of the above hash function variations (several can be set at a given time). If several functions are enabled at the same time, priority is defined as follows (skip functions that are not enabled):

IPv4 packet:

1. Try using the TcpIPv4 function.
2. Try using IPV4_UDP function.
3. Try using the IPv4 function.

IPv6 packet:

1. If TcpIPv6Ex is enabled, try using the TcpIPv6Ex function; else if TcpIPv6 is enabled try using the TcpIPv6 function.
2. If UdpIPv6Ex is enabled, try using UdpIPv6Ex function; else if UpdIPv6 is enabled try using UdpIPv6 function.
3. If IPv6Ex is enabled, try using the IPv6Ex function, else if IPv6 is enabled, try using the IPv6 function.

The following combinations are currently supported:

- Any combination of IPv4, TcpIPv4, and UdpIPv4.
- And/or.
- Any combination of either IPv6, TcpIPv6, and UdpIPv6 or IPv6Ex, TcpIPv6Ex, and UdpIPv6Ex.

When a packet cannot be parsed by the previously mentioned rules, it is assigned an RSS output index = zero. The 32-bit tag (normally a result of the hash function) equals zero.

The 32-bit result of the hash computation is written into the packet descriptor and also provides an index into the indirection table.

The following notation is used to describe the hash functions:

- Ordering is little endian in both bytes and bits. For example, the IP address 161.142.100.80 translates into 0xa18e6450 in the signature.
- A "^ "denotes bit-wise XOR operation of same-width vectors.
- @x-y denotes bytes x through y (including both of them) of the incoming packet, where byte 0 is the first byte of the IP header. In other words, it is considered that all byte-offsets as offsets into a packet where the framing layer header has been stripped out. Therefore, the source IPv4 address is referred to as @12-15, while the destination v4 address is referred to as @16-19.
- @x-y, @v-w denotes concatenation of bytes x-y, followed by bytes v-w, preserving the order in which they occurred in the packet.

All hash function variations (IPv4 and IPv6) follow the same general structure. Specific details for each variation are described in the following section. The hash uses a random secret key length of 320 bits (40 bytes); the key is typically supplied through the RSS Random Key Register (RSSRK).

The algorithm works by examining each bit of the hash input from left to right. Intel's nomenclature defines left and right for a byte-array as follows: Given an array K with k bytes, Intel's nomenclature assumes that the array is laid out as shown:

```
K[0] K[1] K[2] … K[k-1]
```

K[0] is the left-most byte, and the MSB of K[0] is the left-most bit. K[k-1] is the right-most byte, and the LSB of K[k-1] is the right-most bit.

```
ComputeHash(input[], N)
For hash-input input[] of length N bytes (8N bits) and a random secret key
K of 320 bits
Result = 0;
For each bit b in input[] {
if (b == 1) then Result ^= (left-most 32 bits of K);
shift K left 1 bit position;
}
return Result;
```

The following four pseudo-code examples are intended to help clarify exactly how the hash is to be performed in four cases, IPv4 with and without ability to parse the TCP header and IPv6 with an without a TCP header.

26.1.1.8.1.4 Hash for IPv4 with TCP

Concatenate SourceAddress, DestinationAddress, SourcePort, DestinationPort into one single byte-array, preserving the order in which they occurred in the packet:

```
Input[12] = @12-15, @16-19, @20-21, @22-23.
Result = ComputeHash(Input, 12);
```

26.1.1.8.1.5 Hash for IPv4 with UDP

Concatenate SourceAddress, DestinationAddress, SourcePort, DestinationPort into one single byte-array, preserving the order in which they occurred in the packet:

```
Input[12] = @12-15, @16-19, @20-21, @22-23.
Result = ComputeHash(Input, 12);
```

26.1.1.8.1.6 Hash for IPv4 without TCP

Concatenate SourceAddress and DestinationAddress into one single byte-array

```
Input[8] = @12-15, @16-19
Result = ComputeHash(Input, 8)
```

26.1.1.8.1.7 Hash for IPv6 with TCP

Similar to above:

```
Input[36] = @8-23, @24-39, @40-41, @42-43
Result = ComputeHash(Input, 36)
```

26.1.1.8.1.8 Hash for IPv6 with UDP

Similar to above:

```
Input[36] = @8-23, @24-39, @40-41, @42-43
Result = ComputeHash(Input, 36)
```

26.1.1.8.1.9 Hash for IPv6 without TCP

```
Input[32] = @8-23, @24-39
Result = ComputeHash(Input, 32)
```

26.1.1.8.2 Indirection Table

The *RETA* indirection table is a 128-entry structure, indexed by the seven LSB bits of the hash function output. Each entry of the table contains the following:

- Bits [2:0] - RSS index

Note: In RSS only mode, all 3 bits are used. In VMDq mode RSS is not supported.

System software might update the indirection table during run time. Such updates of the table are not synchronized with the arrival time of received packets. Therefore, it is not guaranteed that a table update takes effect on a specific packet boundary.

26.1.1.8.3 RSS Verification Suite

Assume that the random key byte-stream is:

```
0x6d, 0x5a, 0x56, 0xda, 0x25, 0x5b, 0x0e, 0xc2,
0x41, 0x67, 0x25, 0x3d, 0x43, 0xa3, 0x8f, 0xb0,
0xd0, 0xca, 0x2b, 0xcb, 0xae, 0x7b, 0x30, 0xb4,
0x77, 0xcb, 0x2d, 0xa3, 0x80, 0x30, 0xf2, 0x0c,
0x6a, 0x42, 0xb7, 0x3b, 0xbe, 0xac, 0x01, 0xfa
```

26.1.1.8.3.10 IPv4

Table 26-1. IPv4

Destination Address/Port	Source Address/Port	IPv4 Only	IPv4 With TCP
161.142.100.80:1766	66.9.149.187:2794	0x323e8fc2	0x51ccc178
65.69.140.83:4739	199.92.111.2:14230	0xd718262a	0xc626b0ea
12.22.207.184:38024	24.19.198.95:12898	0xd2d0a5de	0x5c2b394a
209.142.163.6:2217	38.27.205.30:48228	0x82989176	0xafc7327f
202.188.127.2:1303	153.39.163.191:44251	0x5d1809c5	0x10e828a2

26.1.1.8.3.11 IPv6

The IPv6 address tuples are only for verification purposes and might not make sense as a tuple.

Table 26-2. IPv6

Destination Address/Port	Source Address/Port	IPv6 Only	IPv6 With TCP
3ffe:2501:200:3::1 (1766)	3ffe:2501:200:1fff::7 (2794)	0x2cc18cd5	0x40207d3d
ff02::1 (4739)	3ffe:501:8::260:97ff:fe40:efab (14230)	0x0f0c461c	0xdde51bbf
fe80::200:f8ff:fe21:67cf (38024)	3ffe:1900:4545:3:200:f8ff:fe21:67cf (44251)	0x4b61e985	0x02d1feef

26.1.1.8.4 Association Through MAC Address

Each of the 24 MAC address filters can be associated with a VM. The *POOLSEL* field in the Receive Address High (RAH) register determines the target VM. Packets that do not match any of the MAC filters (such as promiscuous) are assigned with the default VM as defined in the VT_CTL.DEF_PL field.

Software can program different values to the MAC filters (any bits in RAH or RAL) at any time. The Controller would respond to the change on a packet boundary but does not guarantee the change to take place at some precise time.

26.1.2 L2 Packet Filtering

The receive packet filtering role is to determine which of the incoming packets are allowed to pass to the local system and which of the incoming packets should be dropped since they are not targeted to the local system. Received packets can be destined to the host, to a manageability controller (BMC), or to both. This section describes how host filtering is done, and the interaction with management filtering.

As shown in Figure 26-12, host filtering has three stages:

1. Packets are filtered by L2 filters (MAC address, unicast/multicast/broadcast). See Section 26.1.2.1 for details.
2. Packets are then filtered by VLAN if a VLAN tag is present. See Section 26.1.2.2 for details.
3. Packets are filtered by the manageability filters (port, IP, flex, other). See Section 27.4 for details.

A packet is not forwarded to the host if any of the following takes place:

1. The packet does not pass MAC address filters as described later in this section.

- The packet does not pass VLAN filtering as described later in this section.
- The packet passes manageability filtering and then the manageability filters determine if the packet should be sent only to the BMC (see Section 27.4 and the *MNGONLY* register).

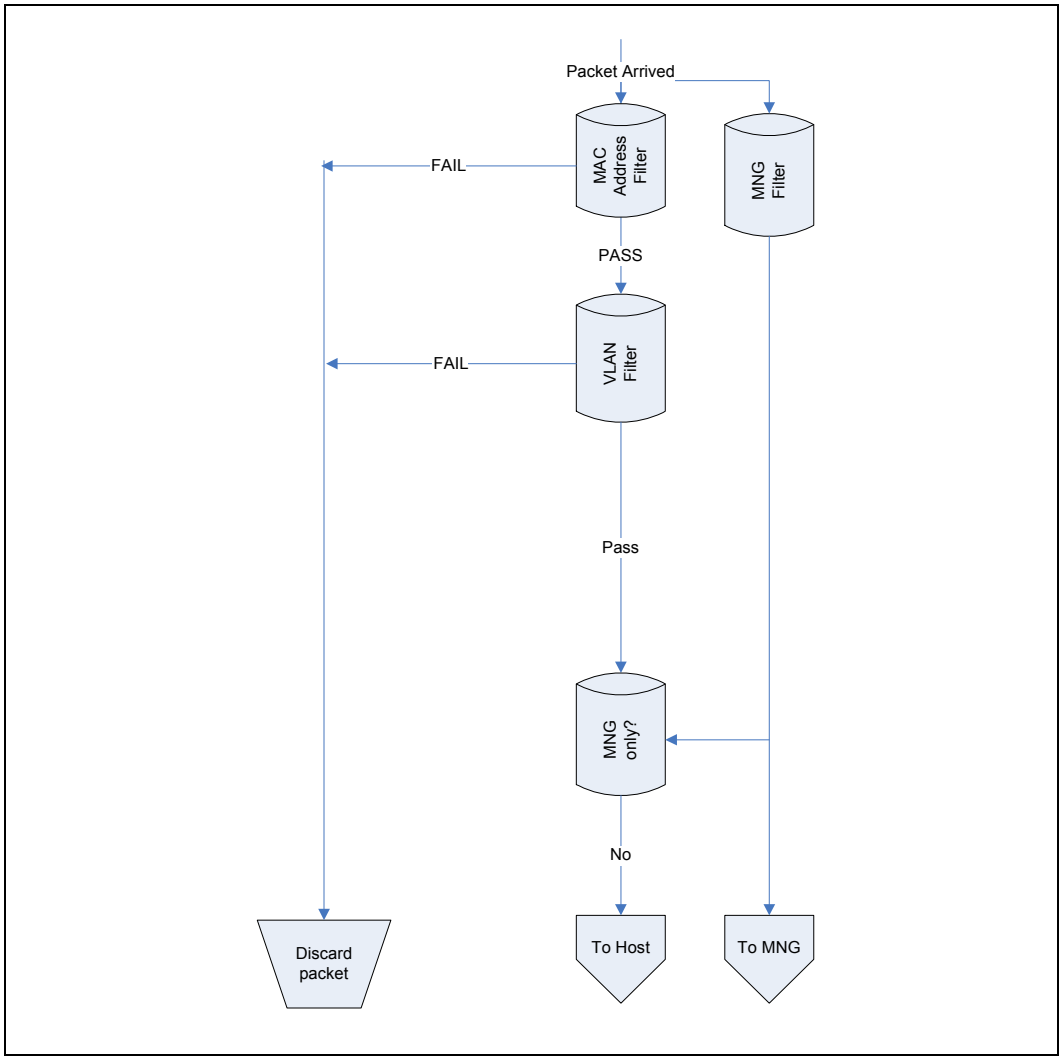
A packet that passes receive filtering as previously described might still be dropped due to other reasons. Normally, only good packets are received. These are defined as those packets with no Under Size Error, Over Size Error, Packet Error, Length Error and CRC Error are detected. However, if the *store-bad-packet* bit is set (*FCTRL.SBP*), then bad packets that don't pass the filter function are stored in host memory. Packet errors are indicated by error bits in the receive descriptor (*RDESC.ERRORS*). It is possible to receive all packets, regardless of whether they are bad, by setting the promiscuous enabled (Unicast and Multicast) and the *store-bad-packet* bits in the *RCTL* register.

If there is insufficient space in the receive FIFO, hardware drops the packet and indicates the missed packet in the appropriate statistics registers.

When the packet is routed to a queue with the *SRRCTL.Drop_En* bit set to 1, receive packets are dropped when insufficient receive descriptors exist to write the packet into system memory.

Note: CRC errors before the SFD are ignored. Any packet must have a valid SFD in order to be recognized by the Controller (even bad packets).

Figure 26-12.Receive Filtering Flow Chart

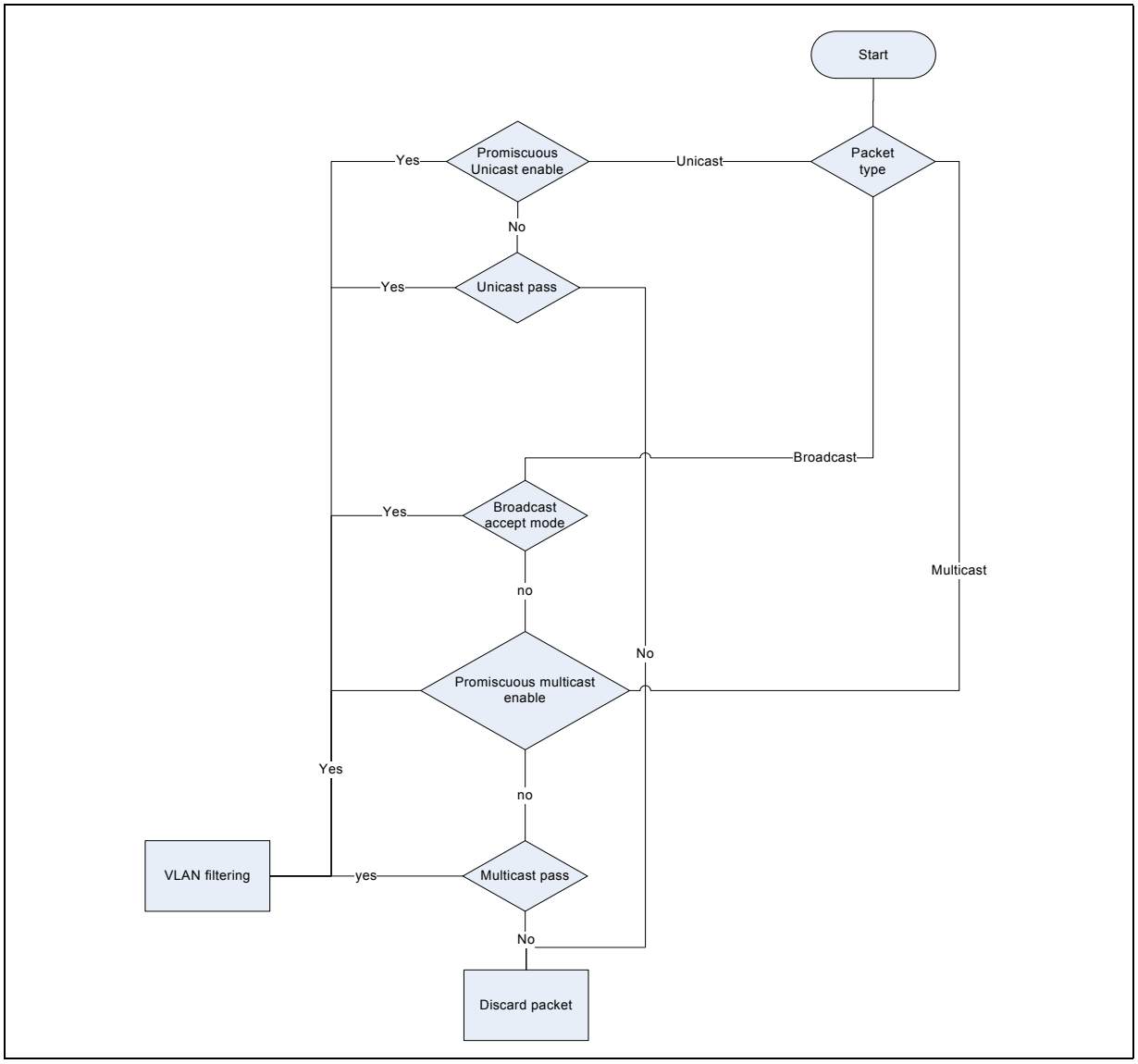


26.1.2.1 MAC Address Filtering

Figure 26-13 shows the MAC address filtering. A packet passes successfully through the MAC address filtering if any of the following conditions are met:

1. It is a unicast packet and promiscuous unicast filtering is enabled.
2. It is a multicast packet and promiscuous multicast filtering is enabled.
3. It is a unicast packet and it matches one of the unicast MAC filters.
4. It is a multicast packet and it matches one of the multicast filters.
5. It is a broadcast packet and Broadcast Accept Mode (RCTL.BAM) is enabled.

Figure 26-13.Host MAC Address Receive Filtering Flow Chart



26.1.2.1.1 Unicast Filter

The entire MAC address is checked against the 24 host unicast addresses. The 24 host unicast addresses are controlled by the host interface (the BMC must not change them). The other 2 addresses are dedicated to management functions and are only accessed by the BMC. The destination address of incoming packet must exactly match one of the pre-configured host address filters. These addresses can be unicast or multicast. Those filters are configured through *RAL*, and *RAH* registers.

Promiscuous Unicast — Receive all unicasts. Promiscuous unicast mode in the *RCTL* register can be set/cleared only through the host interface (not by the BMC). This mode is usually used when the Controller is used as a sniffer.

Unicast Hash Table — Destination address matching the Unicast Hash Table (*UTA*).

26.1.2.1.2 Multicast Filter (Partial)

The 12-bit portion of incoming packet multicast address must exactly match Multicast Filter Address (*MFA*) in order to pass multicast filtering. Those 12 bits out of 48 bits of the destination address can be selected by the *MO* field of *RCTL* (Section 28.9.1.1). These entries can be configured only by the host interface and cannot be controlled by the BMC. Packets received according to this mode have the *PIF* bit in the descriptor set to indicate imperfect filtering that should be validated by the software device driver.

Promiscuous Multicast — Receive all multicast. Promiscuous multicast mode can be set/cleared in the *RCTL* register only through the host interface (not by the BMC) and it is usually used when the Controller is used as a sniffer.

Note: When the promiscuous bit is set and a multicast packet is received, the *PIF* bit of the packet status is not set.

26.1.2.2 VLAN Filtering

A receive packet that successfully passed MAC address filtering is then subjected to VLAN header filtering.

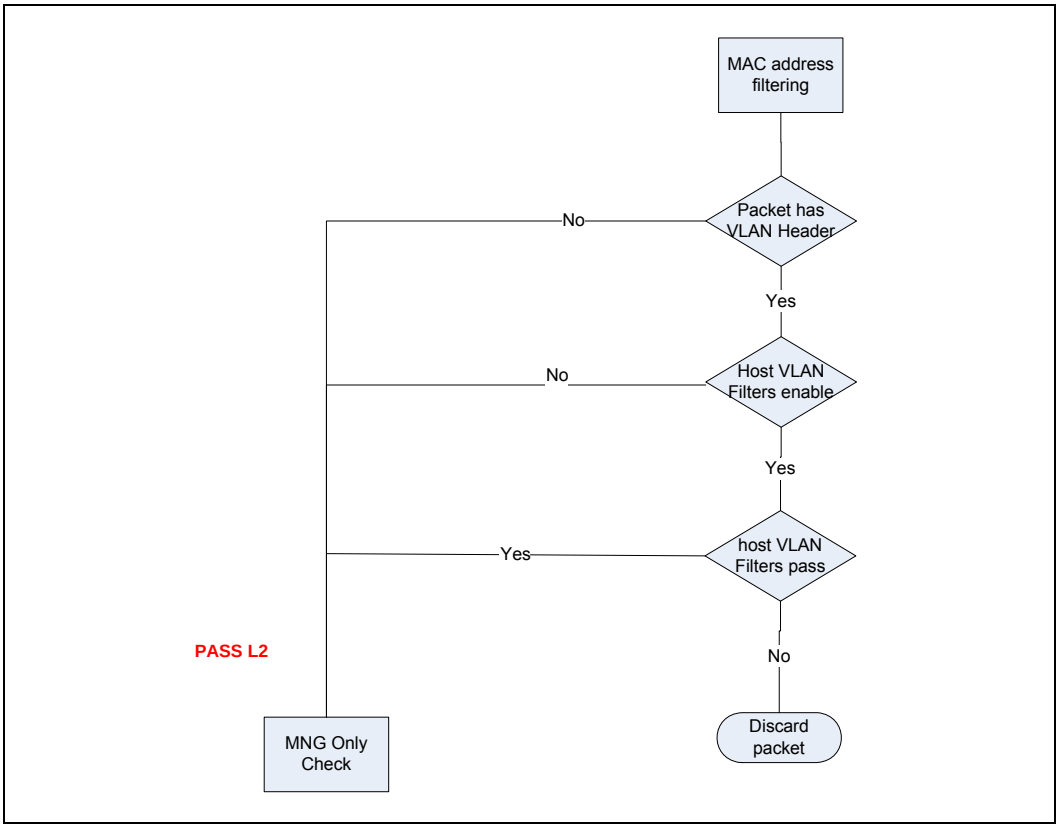
1. If the packet does not have a VLAN header, it passes to the next filtering stage.

Note: If extended VLAN is enabled (*CTRL_EXT.EXTENDED_VLAN* is set), it is assumed that the first VLAN tag is an extended VLAN and it is skipped. All next stages refer to the second VLAN.

2. If VLAN filtering is disabled (*RCTL.VFE* bit is cleared), the packet is forwarded to the next filtering stage.
3. If the packet has a VLAN header, and it matches an enabled host VLAN filter (relevant bit in *VFTA* table is set), the packet is forwarded to the next filtering stage.
4. Otherwise, the packet is dropped.

Figure 26-14 shows the VLAN filtering flow.

Figure 26-14.VLAN Filtering



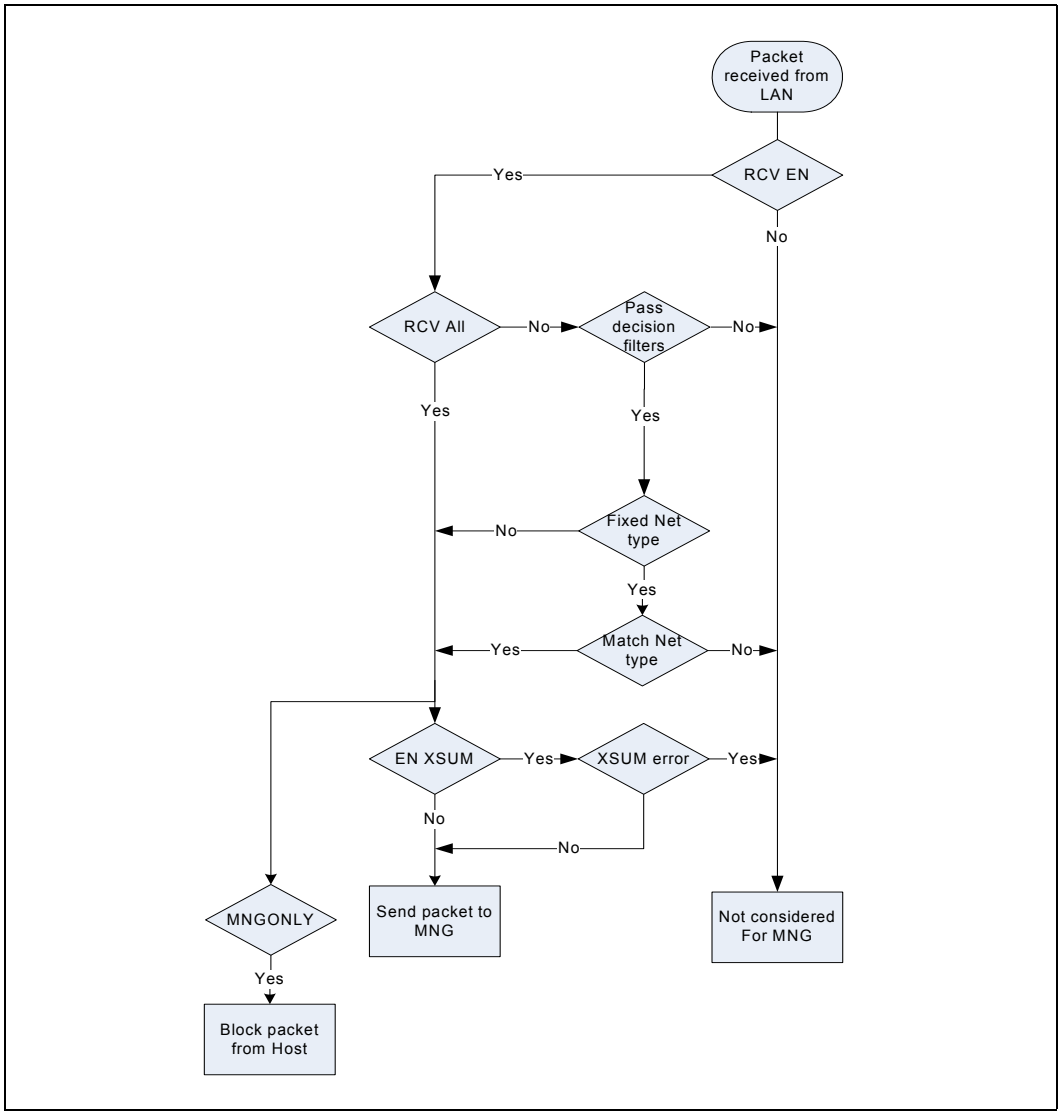
26.1.2.3 Platform Manageability Filtering

Platform manageability filtering is described in Section 27.4.

Figure 26-15 shows the manageability portion of the packet filtering and it is brought here to make the receive packet filtering functionality description complete.

Note: The manageability engine (for example, BMC) might decide to block part of the received packets from also being sent to the Host, according to the external BMC instructions and the EEPROM settings.

Figure 26-15.Platform Manageability Filtering



26.1.3 Receive Data Storage

26.1.3.1 Host Buffers

Each descriptor points to one or more memory buffers that are designated by the software device driver to store packet data.

The size of the buffer can be set using the per queue *SRRCTL[n].BSIZEPACKET* field.

In addition, for advanced descriptor usage the *SRRCTL.BSIZEHEADER* field is used to define the size of the buffers allocated to headers.

The Controller places no alignment restrictions on receive memory buffer addresses. This is desirable in situations where the receive buffer was allocated by higher layers in the networking software stack, as these higher layers might have no knowledge of a specific device's buffer alignment requirements.

Note: When the *No-Snoop Enable* bit is used in advanced descriptors, the buffer address is 16-bit (2-byte) aligned.

26.1.3.2 On-Chip Receive Buffers

The Controller allocates by default a 32 KB on-chip packet buffer per port. The buffer can be used to store packets until they are forwarded to the host. The Controller utilizes a single common ram structure for the on-chip receive buffers allocated to the various ports. If a port is disabled, so that it can't be accessed by host and management, by:

- Setting EEPROM bit *LAN_DIS* or the LAN_PCI_DIS in the "*Software Defined Pins Control*" word for the relevant port to 1.

The freed buffer space can be allocated to the active ports via the "*Initialization Control 4*" EEPROM word. Actual on-chip receive buffer allocated to the port can be read in the *IRPBS register*.

26.1.3.3 On-Chip Descriptor Buffers

The Controller contains a 16 descriptor cache for each receive queue used to reduce the latency of packet processing and to optimize the usage of PCIe* bandwidth by fetching and writing back descriptors in bursts. The fetch and writeback algorithm are described in Section 26.1.6 and Section 26.1.7.

26.1.4 Legacy Receive Descriptor Format

A receive descriptor is a data structure that contains the receive data buffer address and fields for hardware to store packet information. If *SRRCTL[n].DESCTYPE* = 000b, the Controller uses the legacy Receive descriptor as shown in Table 26-3.

Note: Legacy descriptors should not be used when advanced features such as Virtualization are activated.

Table 26-3. Legacy Receive Descriptor (RDESC) Layout

	63	48 47	40 39	32 31 0	16 15
0	Buffer Address [63:0]				
8	VLAN Tag	Errors	Status	Fragment Checksum	Length

After receiving a packet for the Controller, hardware stores the packet data into the indicated buffer and writes the length, packet checksum, status, errors, and status fields.

Packet Buffer Address (64) - Physical address of the packet buffer.

Length Field (16)

Length covers the data written to a receive buffer including CRC bytes (if any). Software must read multiple descriptors to determine the complete length for a packet that spans multiple receive buffers.

Fragment Checksum (16)

This field is used to provide the fragment checksum value. This field equals to the unadjusted 16-bit ones complement of the packet. Checksum calculation starts at the L4 layer (after the IP header) until the end of the packet excluding the CRC bytes. In order to use the fragment checksum assist to offload L4 checksum verification, software might need to back out some of the bytes in the packet. For more details see Section 26.1.11.2

Status Field (8)

Status information indicates whether the descriptor has been used and whether the referenced buffer is the last one for the packet. See Table 26-4 for the layout of the *Status* field. Error status information is shown in Figure 26-8.

Table 26-4. Receive Status (RDESC.STATUS) Layout

7	6	5	4	3	2	1	0
PIF	IPCS	L4CS	UDPCS	VP	Rsv	EOP	DD

- PIF (bit 7) - Passed in-exact filter
- IPCS (bit 6) - Ipv4 checksum calculated on packet
- L4CS (bit 5) - L4 (UDP or TCP) checksum calculated on packet
- UDPCS (bit 4) - UDP checksum calculated on packet
- VP (bit 3) - Packet is 802.1q (matched VET); indicates strip VLAN in 802.1q packet
- RSV (bit 2) - Reserved
- EOP (bit 1) - End of packet
- DD (bit 0) - Descriptor done

EOP and DD

The following table lists the meaning of these bits:

Table 26-5. Receive Status Bits

DD	EOP	Description
0b	0b	Software setting of the descriptor when it hands it off to the hardware.
0b	1b	Reserved (invalid option).
1b	0b	A completion status indication for a non-last descriptor of a packet that spans across multiple descriptors. In a single packet case, DD indicates that the hardware is done with the descriptor and its buffers. Only the *Length* fields are valid on this descriptor.
1b	1b	A completion status indication of the entire packet. Software might take ownership of its descriptors. All fields in the descriptor are valid (reported by the hardware).

VP Field

The *VP* field indicates whether the incoming packet's type matches the VLAN Ethernet Type programmed in the VET Register. For example, if the packet is a VLAN (802.1q) type, it is set if the packet type matches VET and CTRL.VME is set (Vlan mode enabled). It also indicates that VLAN has been stripped from the 802.1q packet. For more details, see Section 26.4.

IPCS (Ipv4 Checksum), L4CS (L4 Checksum), and UDPCS (UDP Checksum)

The meaning of these bits is shown in the table below:

Table 26-6. IPCS, L4CS, and UDPCS

L4CS	UDPCS	IPCS	Functionality
0b	0b	0b	Hardware does not provide checksum offload. Special case: Hardware does not provide UDP checksum offload for IPV4 packet with UDP checksum = 0b
1b	0b	1b / 0b	Hardware provides IPv4 checksum offload if IPCS is active and TCP checksum is offload. A pass/fail indication is provided in the *Error* field – IPE and L4E.
0b	1b	1b / 0b	Hardware provides IPv4 checksum offload if IPCS is active and UDP checksum is offload. A pass/fail indication is provided in the *Error* field – IPE and L4E.

See Table 26-18 for a description of supported packet types for receive checksum offloading. Unsupported packet types do not have the *IPCS* or *L4CS* bits set. IPv6 packets do not have the *IPCS* bit set, but might have the *L4CS* bit set if the Controller recognized the TCP or UDP packet.

PIF

Hardware supplies the *PIF* field to expedite software processing of packets. Software must examine any packet with *PIF* set to determine whether to accept the packet. If *PIF* is clear, then the packet is known to be for this station, so software need not look at the packet contents. Multicast packets passing only the Multicast Vector (MTA) or unicast packets passing only the Unicast Hash Table (UTA) but not any of the MAC address exact filters (RAH, RAL) have *PIF* set. In addition, the following condition causes *PIF* to be cleared:

- The DA of the packet is a multicast address and promiscuous multicast is set (RCTL.MPE = 1b).
- The DA of the packet is a broadcast address and accept broadcast mode is set (RCTL.BAM = 1b)

A MAC control frame forwarded to the host (RCTL.PMCF = 0b) that does not match any of the exact filters, has the *PIF* bit set.

Error Field (8)

Most error information appears only when the *store-bad-packet* bit (RCTL.SBP) is set and a bad packet is received. See Table 26-7 for a definition of the possible errors and their bit positions.

Table 26-7. RXE, IPE and L4E

7	6	5	4	3	2	1	0
RXE	IPE	L4E	Reserved				

- RXE (bit 7) - RX Data Error
- IPE (bit 6) - Ipv4 Checksum Error
- L4E (bit 5) - TCP/UDP Checksum Error
- Reserved (bit 4:0)

IPE/L4E

The IP and TCP/UDP checksum error bits from Table 26-7 are valid only when the IPv4 or TCP/UDP checksum(s) is performed on the received packet as indicated via IPCS and L4CS. These, along with the other error bits, are valid only when the *EOP* and *DD* bits are set in the descriptor.

Note: Receive checksum errors have no affect on packet filtering.

If receive checksum offloading is disabled (RXCSUM.IPOFL and RXCSUM.TUOFL), the *IPE* and *L4E* bits are 0b.

RXE

The RXE error bit is asserted in one of two cases (software might distinguish between these errors by monitoring the respective statistics registers):

1. CRC error is detected. CRC can be a result of reception of /V/ symbol on the TBI interface or assertion of RxERR on the MII/GMII interface or bad EOP or lose of sync during packet reception. Packets with a CRC error are posted to host memory only when *store-bad-packet* bit (*RCTL.SBP*) is set.

VLAN Tag Field (16)

Hardware stores additional information in the receive descriptor for 802.1q packets. If the packet type is 802.1q (determined when a packet matches VET and RCTL.VME = 1b), then the *VLAN Tag* field records the VLAN information and the four-byte VLAN information is stripped from the packet data storage. Otherwise, the *VLAN Tag* field contains 0x0000. The rule for VLAN tag is to use network ordering (also called big endian). It appears in the following manner in the descriptor:

Table 26-8. VLAN Tag Field Layout (for 802.1q Packet)

15 13	12	11 0
PRI	CFI	VLAN

26.1.5 Advanced Receive Descriptors

26.1.5.1 Advanced Receive Descriptors (RDESC) - Read Format

Table 26-9 shows the receive descriptor. This is the format that software writes to the descriptor queue and hardware reads from the descriptor queue in host memory. Hardware writes back the descriptor in a different format, shown in Table 26-10.

Table 26-9. RDESC Descriptor Read Format

	63 1	0
0	Packet Buffer Address [63:1]	A0/NSE
8	Header Buffer Address [63:1]	DD

Packet Buffer Address (64) - Physical address of the packet buffer. The lowest bit is either A0 (LSB of address) or NSE (No-Snoop Enable), depending on bit *RXCTL.RXdataWriteNSEn* of the relevant queue. See Section 28.12.1.1.

Header Buffer Address (64) - Physical address of the header buffer. The lowest bit is DD.

Note: The Controller does not support null descriptors (a descriptor with a packet or header address that is always equal to zero).

When software sets the *NSE* bit in the receive descriptor, the Controller places the received packet associated with this descriptor in memory at the packet buffer address with *NSE* set in the PCIe* attribute fields. NSE does not affect the data written to the header buffer address.

When a packet spans more than one descriptor, the header buffer address is not used for the second, third, etc. descriptors; only the packet buffer address is used in this case.

NSE is enabled for packet buffers that the software device driver knows have not been touched by the processor since the last time they were used, so the data cannot be in the processor cache and snoop is always a miss. Avoiding these snoop misses improves system performance. No-snoop is particularly useful when the DMA engine is moving the data from the packet buffer into application buffers, and the software device driver is using the information in the header buffer for its work with the packet.

Note: When No-Snoop Enable is used, relaxed ordering should also be enabled with *CTRL_EXT.RO_DIS*.

26.1.5.2 Advanced Receive Descriptors (RDESC) - Writeback Format

When the Controller writes back the descriptors, it uses the descriptor format shown in Table 26-10.

Note: SRRCTL[n]. DESCTYPE must be set to a value other than 000b for the Controller to write back the special descriptors.

Table 26-10. RDESC Descriptor Write-Back Format

	63 48	47 35	34 32	31	30 21	20	19 17	16 4	3 0
0	RSS Hash Value/Fragment Checksum and IP identification			SPH	HDR_LEN	RSV		Packet Type	RSS Type
8	VLAN Tag	PKT_LEN		Extended Error		Extended Status			

RSS Type (4)

Table 26-11. RSS Type

Packet Type	Description
0x0	No hash computation done for this packet.
0x1	HASH_TCP_IPV4
0x2	HASH_IPV4
0x3	HASH_TCP_IPV6
0x4	HASH_IPV6_EX
0x5	HASH_IPV6
0x6	HASH_TCP_IPV6_EX
0x7	HASH_UDP_IPV4
0x8	HASH_UDP_IPV6
0x9	HASH_UDP_IPV6_EX
0xA:0xF	Reserved

The Controller must identify the packet type and then choose the appropriate RSS hash function to be used on the packet. The RSS type reports the packet type that was used for the RSS hash function.

Packet Type (13)

- VPKT (bit 12) - VLAN Packet indication

The 12 LSB bits of the packet type reports the packet type identified by the hardware as follows:

Table 26-12. Packet Type LSB Bits (11:10)

Bit Index	Bit 11 = 0b	Bit 11 = 1b (L2 packet)
0	IPV4 - IPv4 header present	EtherType - ETQF register index that matches the packet. Special types might be defined for 1588, 802.1X, LLDP or any other requested type.
1	IPV4E - IPv4 Header includes extensions	
2	IPV6 - IPv6 header present	
3	IPV6E- IPv6 Header includes extensions	Reserved
4	TCP - TCP header present	
5	UDP - UDP header present	Reserved
6	SCTP - SCTP header present	Reserved
7	NFS - NFS header present	Reserved
10:8	Reserved	Reserved

RSV(5): Reserved.

HDR_LEN (10) - The length (bytes) of the header as parsed by the Controller. In split mode when HBO (Header Buffer Overflow) is set in the Extended error field, the HDR_LEN can be greater then zero though nothing is written to the header buffer. In header replication mode, the HDR_LEN field does not reflect the size of the data actually stored in the header buffer because the Controller fills the buffer up to the size configured by SRRCTL[n].BSIZEHEADER, which might be larger than the header size reported here. This field is only valid in the first descriptor of a packet and should be ignored in all subsequent descriptors.

Note: When the packet is time stamped and the time stamp is placed at the beginning of the buffer the *RDESC.HDR_LEN* field is updated with the additional time stamp bytes (16 bytes). For further information see Section 26.1.10.

Other packet types are posted sequentially in the host packet buffer. Each line in the following table has an enable bit in the PSRTYPE register. When one of the bits is set, the corresponding packet type is split. If the bit is not set, a packet matching the header layout is not split.

Header split and replication is described in Section 26.1.9 while the packet types for this functionality are enabled by the PSRTYPE[n] registers (Section 28.9.1.3).

Note: The header of a fragmented IPv6 packet is defined before the fragmented extension header.

SPH (1) - Split Header - When set, indicates that the *HDR_LEN* field reflects the length of the header found by hardware. If cleared, the *HDR_LEN* field should be ignored, unless *SRRCTL[n].DESCTYPE* is set to *Split – always use header buffer* mode and *PKT_LEN* = 0. In this case, the *HDR_LEN* reflects the size of the packet, even if *SPH* bit is cleared.

In the case were *SRRCTL[n].DESCTYPE* is set to *Header replication mode*, *SPH* bit is set but the *HDR_LEN* field does not reflect the size of the data actually stored in the header buffer, because the Controller fills the buffer up to the size configured by *SRRCTL[n].BSIZEHEADER*.

RSS Hash / Fragment Checksum (32)

This field has multiplexed functionality according to the received packet type (reported on the *Packet Type* field in this descriptor) and device setting.

Fragment Checksum (16-Bit; 63:48)

The fragment checksum word contains the unadjusted one's complement checksum of the IP payload and is used to offload checksum verification for fragmented UDP packets as described in Section 26.1.11.2. This field is mutually exclusive with the RSS hash. It is enabled when the *RXCSUM.PCSD* bit is cleared and the *RXCSUM.IPPCSE* bit is set.

IP identification (16-Bit; 47:32)

The IP identification word identifies the IP packet to whom this fragment belongs and is used to offload checksum verification for fragmented UDP packets as described in Section 26.1.11.2. This field is mutually exclusive with the RSS hash. It is enabled when the *RXCSUM.PCSD* bit is cleared and the *RXCSUM.IPPCSE* bit is set.

RSS Hash Value (32)

The RSS hash value is required for RSS functionality as described in Section 26.1.1.8. This bit is mutually exclusive with the fragment checksum. It is enabled when the *RXCSUM.PCSD* bit is set.

Extended Status (20)

Status information indicates whether the descriptor has been used and whether the referenced buffer is the last one for the packet. Table 26-13 lists the extended status word in the last descriptor of a packet (*EOP* is set). Table 26-14 lists the extended status word in any descriptor but the last one of a packet (*EOP* is cleared).

Table 26-13. Receive Status (RDESC.STATUS) Layout of Last Descriptor

19	18	17	16	15	14	13	12	11	10
	Rsv		TS	TSIP	Reserved		Strip CRC	LLINT	UDPV

VEXT	Rsv	PIF	IPCS	L4I	UDPCS	VP	Rsv	EOP	DD
9	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Table 26-14. Receive Status (RDESC.STATUS) Layout of Non-Last Descriptor

19	2	1	0
Reserved		EOP = 0b	DD

TS (16) - Time Stamped Packet (Time Sync). The Time Stamp bit is set to indicate that the device recognized a Time Sync packet and time stamped it in the *RXSTMPL/H* time stamp registers (See Section 26.9.2.2 and Section 26.9.1.1).

TSIP (15) - Timestamp in packet. The Timestamp In Packet bit is set to indicate that the received packet arrival time was captured by the hardware and the timestamp was placed in the receive buffer. For further details see Section 26.1.10.

Reserved (2, 8, 14:13, 18) - Reserved at zero.

PIF (7), IPCS(6), UDPCS(4), VP(3), EOP (1), DD (0) - These bits are described in the legacy descriptor format in Section 26.1.4.

L4I (5) - This bit indicates that an L4 integrity check was done on the packet, either TCP checksum, UDP checksum or SCTP CRC checksum. This bit is valid only for the last descriptor of the packet. An error in the integrity check is indicated by the *L4E* bit in the error field. The type of check done can be induced from the packet type bits 4, 5 and 6. If bit 4 is set, a TCP checksum was done. If bit 5 is set a UDP checksum was done, and if bit 6 is set, a SCTP CRC checksum was done.

VEXT (9) - First VLAN is found on a double VLAN packet. This bit is valid only when *CTRL_EXT.EXTENDED_VLAN* is set. For more details see Section 26.4.5.

UDPV (10) - This bit indicates that the incoming packet contains a valid (non-zero value) checksum field in an incoming fragmented UDP Ipv4 packet. This means that the *Fragment Checksum* field in the receive descriptor contains the UDP checksum as described in Section 26.1.11.2. When this field is cleared in the first fragment that contains the UDP header, means that the packet does not contain a valid UDP checksum and the checksum field in the Rx descriptor should be ignored. This field is always cleared in incoming fragments that do not contain the UDP header.

LLINT (11) - This bit indicates that the packet caused an immediate interrupt via the low latency interrupt mechanism.

Strip CRC (12) - This bit indicates that Ethernet CRC has been stripped from incoming packet. Strip CRC operation is defined by the *RCTL.SECRC* bit.

Extended Error (12)

Table 26-15 and the text that follows describes the possible errors reported by hardware

Table 26-15. Receive Errors (RDESC.ERRORS) Layout

11	10	9	8 7	6 4	3	2 0
RXE	IPE	L4E		Reserved	HBO	Reserved

RXE (bit 11) - RXE is described in the legacy descriptor format in Section 26.1.4.

IPE (bit 10) - The IPE error indication is described in the legacy descriptor format in Section 26.1.4.

L4E (bit 9) - L4 error indication - When set, indicates that hardware attempted to do an L4 integrity check as described in the *L4I* bit, but the check failed.

Reserved (bits 6:4)

HBO (bit 3) - Header Buffer Overflow

Note: The HBO bit is relevant only if *SPH* is set.

1. In both header replication modes, *HBO* is set if the header size (as calculated by hardware) is bigger than the allocated buffer size (*SRRCTL.BSIZEHEADER*) but the replication still takes place up to the header buffer size. Hardware sets this bit in order to indicate to software that it needs to allocate bigger buffers for the headers.
2. In header split mode, when *SRRCTL[n] BSIZEHEADER* is smaller than *HDR_LEN*, then *HBO* is set to 1b, In this case, the header is not split. Instead, the header resides within the host packet buffer. The *HDR_LEN* field is still valid and equal to the calculated size of the header. However, the header is not copied into the header buffer.
3. In header split mode, always use header buffer mode, when *SRRCTL[n] BSIZEHEADER* is smaller than *HDR_LEN*, then *HBO* is set to 1b. In this case, the header buffer is used as part of the data buffers and contains the first

BSIZEHEADER bytes of the packet. The *HDR_LEN* field is still valid and equal to the calculated size of the header.

Note: Most error information appears only when the *store–bad–packet* bit (*RCTL.SBP*) is set and a bad packet is received.

Reserved (bits 2:0) - Reserved

PKT_LEN (16) – Number of bytes existing in the host packet buffer

The length covers the data written to a receive buffer including CRC bytes (if any). Software must read multiple descriptors to determine the complete length for packets that span multiple receive buffers. If SRRCTL.DESC_TYPE = 4 (advanced descriptor header replication large packet only) and the total packet length is smaller than the size of the header buffer (no replication is done), this field continues to reflect the size of the packet, although no data is written to the packet buffer. Otherwise, if the buffer is not split because the header is bigger than the allocated header buffer, this field reflects the size of the data written to the first packet buffer (header and data).

Note: When the packet is time stamped and the time stamp is placed at the beginning of the buffer, the *RDESC.PKT_LEN* field is updated with the additional time stamp bytes (16 bytes). For further information see Section 26.1.10.

VLAN Tag (16)

These bits are described in the legacy descriptor format in Section 26.1.4.

26.1.6 Receive Descriptor Fetching

The fetching algorithm attempts to make the best use of PCIe* bandwidth by fetching a cache-line (or more) descriptor with each burst. The following paragraphs briefly describe the descriptor fetch algorithm and the software control provided.

When the on-chip buffer is empty, a fetch happens as soon as any descriptors are made available (host writes to the tail pointer). When the on-chip buffer is nearly empty (RXDCTL.PTHRESH), a prefetch is performed each time enough valid descriptors (RXDCTL.HTHRESH) are available in host memory.

When the number of descriptors in host memory is greater than the available on-chip descriptor cache, the Controller might elect to perform a fetch that is not a multiple of cache-line size. Hardware performs this non-aligned fetch if doing so results in the next descriptor fetch being aligned on a cache-line boundary. This enables the descriptor fetch mechanism to be most efficient in the cases where it has fallen behind software.

All fetch decisions are based on the number of descriptors available and do not take into account any split of the transaction due to bus access limitations.

Note: The Controller NEVER fetches descriptors beyond the descriptor tail pointer.

26.1.7 Receive Descriptor Write-Back

Processors have cache-line sizes that are larger than the receive descriptor size (16 bytes). Consequently, writing back descriptor information for each received packet would cause expensive partial cache-line updates. A receive descriptor packing mechanism minimizes the occurrence of partial line write-backs.

To maximize memory efficiency, receive descriptors are packed together and written as a cache-line whenever possible. Descriptors write-backs accumulate and are opportunistically written out in cache line-oriented chunks, under the following scenarios:

- RXDCTL.WTHRESH descriptors have been used (the specified maximum threshold of unwritten used descriptors has been reached).
- The receive timer expires (EITR) - in this case all descriptors are flushed ignoring any cache-line boundaries.
- Explicit software flush (*RXDCTL.SWFLS*).
- Dynamic packets - if at least one of the descriptors that are waiting for write-back are classified as packets requiring immediate notification the entire queue is flushed out.

When the number of descriptors specified by RXDCTL.WTHRESH have been used, they are written back regardless of cache-line alignment. It is therefore recommended that WTHRESH be a multiple of cache-line size. When the receive timer (EITR) expires, all used descriptors are forced to be written back prior to initiating the interrupt, for consistency. Software might explicitly flush accumulated descriptors by writing the RXDCTLn register with the *SWFLS* bit set.

When the Controller does a partial cache-line write-back, it attempts to recover to cache-line alignment on the next write-back.

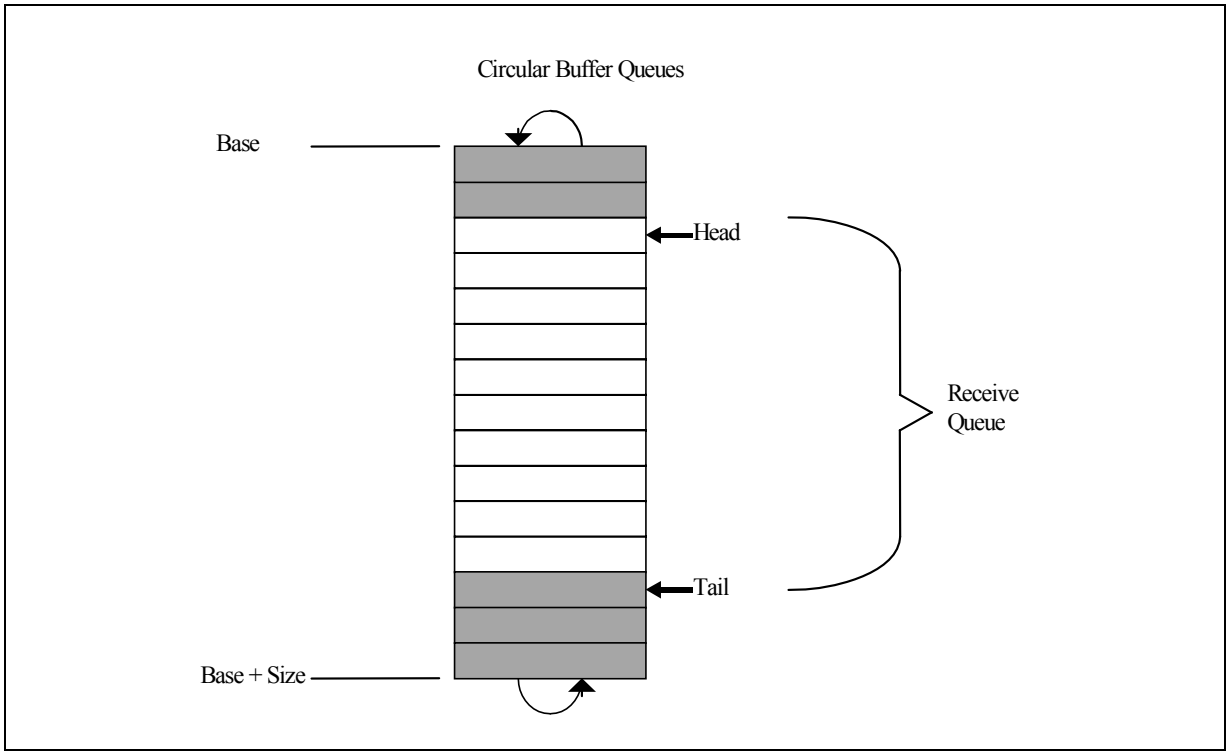
For applications where the latency of received packets is more important that the bus efficiency and the CPU utilization, an EITR value of zero may be used. In this case, each receive descriptor will be written to the host immediately. If RXDCTL.WTHRESH equals zero, then each descriptor will be written back separately, otherwise, write back of descriptors may be coalesced if descriptor accumulates in the internal descriptor ring due to bandwidth constraints.

All write-back decisions are based on the number of descriptors available and do not take into account any split of the transaction due to bus access limitations.

26.1.8 Receive Descriptor Ring Structure

Figure 26-16 shows the structure of each of the 16 receive descriptor rings. Hardware maintains 16 circular queues of descriptors and writes back used descriptors just prior to advancing the head pointer(s). Head and tail pointers wrap back to base when size descriptors have been processed.

Figure 26-16.Receive Descriptor Ring Structure



Software inserts receive descriptors by advancing the tail pointer(s) to refer to the address of the entry just beyond the last valid descriptor. This is accomplished by writing the descriptor tail register(s) with the offset of the entry beyond the last valid descriptor. The hardware adjusts its internal tail pointer(s) accordingly. As packets arrive, they are stored in memory and the head pointer(s) is incremented by hardware. When the head pointer(s) is equal to the tail pointer(s), the queue(s) is empty. Hardware stops storing packets in system memory until software advances the tail pointer(s), making more receive buffers available.

The receive descriptor head and tail pointers reference to 16-byte blocks of memory. Shaded boxes in Figure 26-16 represent descriptors that have stored incoming packets but have not yet been recognized by software. Software can determine if a receive buffer is valid by reading the descriptors in memory. Any descriptor with a non-zero DD value has been processed by the hardware and is ready to be handled by the software.

Note: The head pointer points to the next descriptor that is written back. After the descriptor write-back operation completes, this pointer is incremented by the number of descriptors written back. Hardware owns all descriptors between [head... tail]. Any descriptor not in this range is owned by software.

The receive descriptor rings are described by the following registers:

- Receive Descriptor Base Address (RDBA7 to RDBA0) register:
  This register indicates the start of the descriptor ring buffer. This 64-bit address is aligned on a 16-byte boundary and is stored in two consecutive 32-bit registers. Hardware ignores the lower 4 bits.
- Receive Descriptor Length (RDLEN7 to RDLEN0) registers:
  This register determines the number of bytes allocated to the circular buffer. This value must be a multiple of 128 (the maximum cache-line size). Since each

descriptor is 16 bytes in length, the total number of receive descriptors is always a multiple of eight.

- Receive Descriptor Head (RDH7 to RDH0) registers:
  This register holds a value that is an offset from the base and indicates the in-progress descriptor. There can be up to 64 KB, 8 KB descriptors in the circular buffer. Hardware maintains a shadow copy that includes those descriptors completed but not yet stored in memory.
- Receive Descriptor Tail (RDT7 to RDT0) registers:
  This register holds a value that is an offset from the base and identifies the location beyond the last descriptor hardware can process. This is the location where software writes the first new descriptor.

If software statically allocates buffers, uses legacy receive descriptors, and uses memory read to check for completed descriptors, it has to zero the status byte in the descriptor before bumping the tail pointer to make it ready for reuse by hardware. Zeroing the status byte is not a hardware requirement but is necessary for performing an in-memory scan.

All the registers controlling the descriptor rings behavior should be set before receive is enabled, apart from the tail registers that are used during the regular flow of data.

26.1.8.1 Low Receive Descriptors Threshold

As described above, the size of the receive queues is measured by the number of receive descriptor. During run time the software processes completed descriptors and then increments the Receive Descriptor Tail registers (*RDT*). At the same time, the hardware may post new packets received from the LAN incrementing the Receive Descriptor Head registers (*RDH*) for each used descriptor.

The number of usable (free) descriptors for the hardware is the distance between Tail and Head registers. When the Tail reaches the Head, there are no free descriptors and further packets may be either dropped or block the receive FIFO. In order to avoid this behavior, the Controller may generate a low latency interrupt (associated with the relevant receive queue) once the amount of free descriptors is less or equal than the threshold. The threshold is defined in 16 descriptors granularity per queue in the *SRRCTL[n].RDMTS* field.

26.1.9 Header Splitting and Replication

26.1.9.1 Purpose

This feature consists of splitting or replicating packet's header to a different memory space. This helps the host to fetch headers only for processing: headers are replicated through a regular snoop transaction in order to be processed by the host CPU. It is recommended to perform this transaction with the DCA feature enabled (see Section 28.12) or in conjunction with a software-prefetch.

The packet (header and payload) is stored in memory through a (optionally) non-snoop transaction. Later, a transaction moves the payload from the software device driver buffer to application memory or it is moved using a normal memory copy operation.

The Controller supports header splitting in several modes:

- Legacy mode: legacy descriptors are used; headers and payloads are not split.
- Advanced mode, no split: advanced descriptors are in use; header and payload are not split.

- Advanced mode, split: advanced descriptors are in use; header and payload are split to different buffers. If the packet cannot be split, only the packet buffer is used.
- Advanced mode, replication: advanced descriptors are in use; header is replicated in a separate buffer and also in a payload buffer.
- Advanced mode, replication, conditioned by packet size: advanced descriptors are in use; replication is performed only if the packet is larger than the header buffer size.
- Advanced mode, split, always use header buffer: advanced descriptors are in use; header and payload are split to different buffers. If no split is done, the first part of the packet is stored in the header buffer.

26.1.9.2 Description

Figure 26-17.Header Splitting

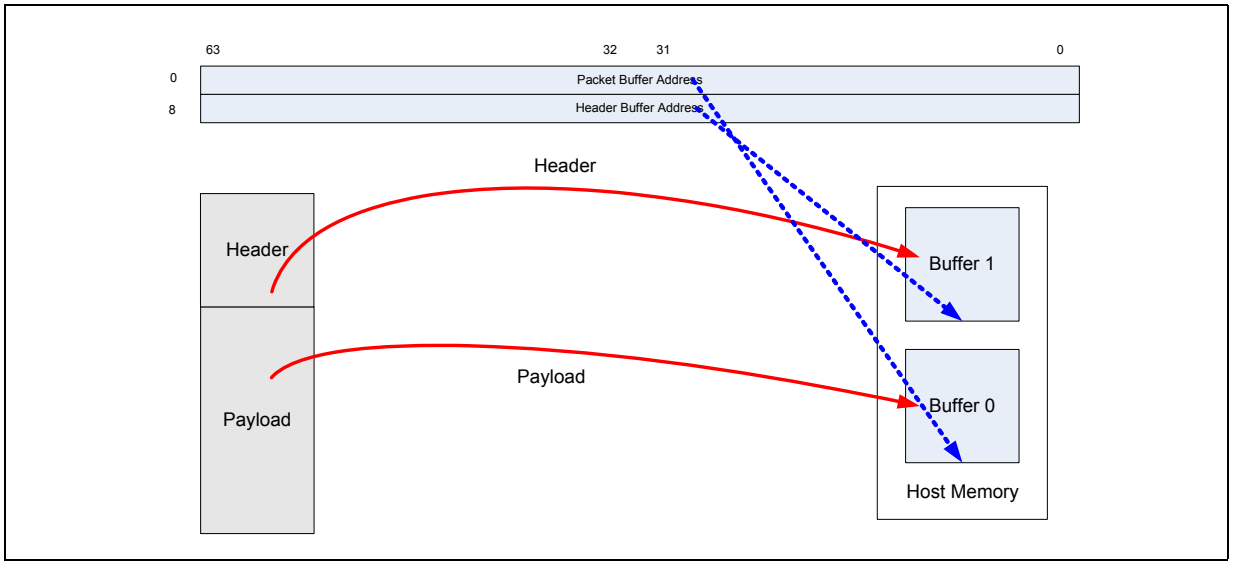
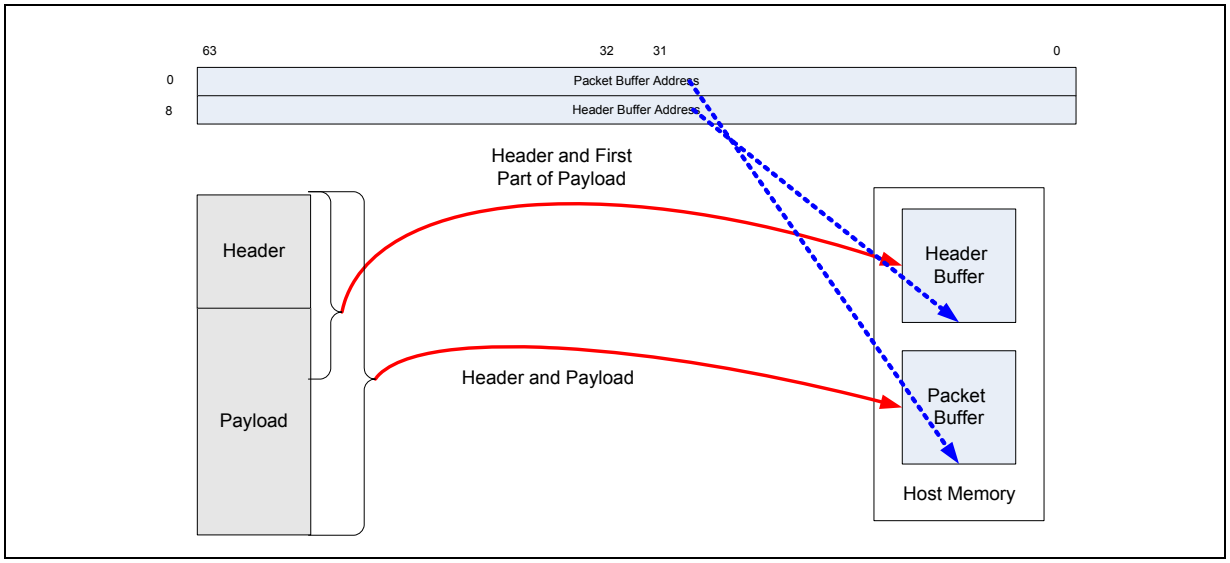


Figure 26-18.Header Replication



The physical address of each buffer is written in the *Buffer Addresses* fields. The sizes of these buffers are statically defined by *BSIZEPACKET* and *BSIZEHEADER* fields in the *SRRCTL[n]* registers.

The packet buffer address includes the address of the buffer assigned to the replicated packet, including header and data payload portions of the received packet. In the case of a split header, only the payload is included.

The header buffer address includes the address of the buffer that contains the header information. The receive DMA module stores the header portion of the received packets into this buffer.

The Controller uses the packet replication or splitting feature when the SRRCTL[n].DESCTYPE is larger than one. The software device driver must also program the buffer sizes in the SRRCTL[n] registers.

When header split is selected, the packet is split only on selected types of packets. A bit exists for each option in PSRTYPE[n] registers so several options can be used in conjunction with them. If one or more bits are set, the splitting is performed for the corresponding packet type.

The following table lists the behavior of the Controller in the different modes

Table 26-16. GBE Controller Split/Replicated Header Behavior

DESCTYPE	Condition	SPH	HBO	PKT_LEN	HDR_LEN	Header and Payload DMA
Split	1. Header can't be decoded	0b	0b	Min(Packet length, Buffer size)	N/A	Header + Payload ‹ Packet buffer
	2. Header <= BSIZEHEADER	1b	0b	Min(Payload length, Buffer size)[1]	Header size	Header ‹ Header buffer Payload ‹ Packet buffer
	3. Header > BSIZEHEADER	1b	1b	Min(Packet length, Buffer size)	Header size[2]	Header + Payload ‹ Packet buffer

Table 26-16. GBE Controller Split/Replicated Header Behavior

Split – always use header buffer	1. Packet length <= BSIZEHEADER	0b	0b	Zero	Packet length	Header + Payload ‹ Header buffer
	2. Header can't be Decoded and Packet length > BSIZEHEADER	0b	0b	Min(Packet length – BSIZEHEADER, Data Buffer size)	BSIZEHEADER	Header + Payload ‹ Header + Packet buffers[3]
	3. Header <= BSIZEHEADER and Packet length >= BSIZEHEADER	1b	0b	Min(Payload length, Data Buffer size)	Header Size	Header ‹ Header buffer Payload ‹ Packet buffer
	4. Header > BSIZEHEADER	1b	1b	Min(Packet length – BSIZEHEADER, Data Buffer size)	Header Size[2]	Header + Payload ‹ Header + Packet buffer[3]
Replicate Large Packet only	1. Header + Payload <= BSIZEHEADER	0b/1b[4]	0b	Packet length	Header size, N/A[4]	Header + Payload ‹ Header buffer
	2. Header + Payload > BSIZEHEADER	0b/1b[4]	0b/1b[5]	Min(Packet length, Buffer size)	Header size, N/A[4]	(Header + Payload)(partial[6]) ‹ Header buffer Header + Payload ‹ Packet buffer

1. In a header only packet (such as TCP ACK packet), the PKT_LEN is zero.
2. The HDR_LEN doesn't reflect the actual data size stored in the Header buffer. It reflects the header size determined by the parser. When timestamp in packet is enabled header size reflects the additional 16 bytes of the timestamp.
3. If the packet spans more than one descriptor, only the header buffer of the first descriptor is used. The header buffer is used for the first part of the packet until it is filled up, and then the first packet buffer is used for the continuation of the packet.

Software Notes:

- If *SRRCTL[n].NSE* is set, all buffers' addresses in a packet descriptor must be word aligned.
- Packet header can't span across buffers, therefore, the size of the header buffer must be larger than any expected header size. Otherwise, only the part of the header fitting the header buffer is replicated. In the case of header split mode (*SRRCTL[n].DESCTYPE* = 010b), a packet with a header larger than the header buffer is not split.

26.1.10 Receive Packet Timestamp in Buffer

The Controller supports adding an optional tailored header before the MAC header of the packet in the receive buffer. The 64 MSB bits of the 128 bit tailored header include a timestamp composed of the packet reception time measured in the SYSTIML (Low DW) and SYSTIMH (High DW) registers (See Section 26.9.2.1 for further information on SYSTIML/H operation). The 64 LSB bits of the tailored header are reserved.

The timestamp information is placed in Networking order (Big Endian) format as can be seen in Table 26-17.

Table 26-17. Timestamp Layout in Buffer

0 3	4 7	8 11	12 15	16...
SYTIMH	STIML	Reserved (0x0)	Reserved (0x0)	Received Packet

When *SRRCTL[n].Timestamp* is set to 1, packets received to the queue will be time stamped if they meet one of the following conditions:

— Meet the criteria defined in the *TSYNCRXCTL.Type* field (See Section 28.16.1.1 and Section 28.16.1.26).

— Match the value defined in one of the *ETQF* registers with the *1588 time stamp* bit set (See Section 26.1.1.4).

— Match a 2-tuple filter with the *TTQF.1588 time stamp* set (See Section 26.1.1.5).

When detecting a receive packet that should be time stamped, the Controller will:

- Place a 64 bit timestamp, indicating the time a packet was received by the MAC, at the beginning of the receive buffer before the received packet.
- Set the *TSIP* bit in the *RDESC.STATUS* field of the last receive descriptor.
- Update the *RDESC.Packet Type* field in the last receive descriptor. Value in this field enables identifying that this is a PTP (Precision Time Protocol) packet (this indication is only relevant for L2 packets).
- Update the *RDESC.HDR_LEN* and *RDESC.PKT_LEN* values to include size of timestamp.

Software driver should take into account the additional size of the timestamp when preparing the receive descriptors for the relevant queue.

26.1.11 Receive Packet Checksum Off Loading

The Controller supports the off loading of three receive checksum calculations: the packet checksum, the IPv4 header checksum, and the TCP/UDP checksum.

The packet checksum is the one's complement over the receive packet, starting from the byte indicated by *RXCSUM.PCSS* (zero corresponds to the first byte of the packet), after stripping. For packets with a VLAN header, the packet checksum includes the header if VLAN striping is not enabled by the *CTRL.VME*. If a VLAN header strip is enabled, the packet checksum and the starting offset of the packet checksum exclude the VLAN header due to masking of VLAN header. For example, for an Ethernet II frame encapsulated as an 802.3ac VLAN packet and *CTRL.VME* is set and with *RXCSUM.PCSS* set to 14, the packet checksum would include the entire encapsulated frame, excluding the 14-byte Ethernet header (DA, SA, type/length) and the 4-byte q-tag. The packet checksum does not include the Ethernet CRC if the *RCTL.SECRC* bit is set.

Software must make the required offsetting computation (to remove the bytes that should not have been included and to include the pseudo-header) prior to comparing the packet checksum against the TCP checksum stored in the packet.

For supported packet/frame types, the entire checksum calculation can be off loaded to the Controller. If *RXCSUM.IPOFL* is set to 1b, the Controller calculates the IPv4 checksum and indicates a pass/fail indication to software via the IPv4 *Checksum Error* bit (*RDESC.IPE*) in the *Error* field of the receive descriptor. Similarly, if *RXCSUM.TUOFL* is set to 1b, the Controller calculates the TCP or UDP checksum and indicates a pass/fail condition to software via the TCP/UDP *Checksum Error* bit (*RDESC.L4E*). These error bits are valid when the respective status bits indicate the checksum was calculated for the packet (*RDESC.IPCS* and *RDESC.L4CS*, respectively). Similarly, if *RFCTL.Ipv6_DIS* and *RFCTL.IP6Xsum_DIS* are cleared to 0b and *RXCSUM.TUOFL* is set to 1b, the Controller calculates the TCP or UDP checksum for IPv6 packets. It then indicates a pass/fail condition in the TCP/UDP *Checksum Error* bit (*RDESC.L4E*).

If neither *RXCSUM.IPOFL* nor *RXCSUM.TUOFL* are set, the *Checksum Error* bits (*IPE* and **L4E**) are 0b for all packets.

Supported frame types:

- Ethernet II
- Ethernet SNAP

Table 26-18. Supported Receive Checksum Capabilities

Packet Type	Hardware IP Checksum Calculation	Hardware TCP/UDP Checksum Calculation
IPv4 packets.	Yes	Yes
IPv6 packets.	No (n/a)	Yes
IPv6 packet with next header options: • Hop-by-hop options • Destinations options (without Home option) • Destinations options (with Home option) • Routing (with Segments Left zero) • Routing (with Segments Left > zero) • Fragment	 No (n/a) No (n/a) No (n/a) No (n/a) No (n/a) No (n/a)	 Yes Yes No Yes No No
IPv4 tunnels: • IPv4 packet in an IPv4 tunnel. • IPv6 packet in an IPv4 tunnel.	 No Yes (IPv4)	 No Yes[1]
IPv6 tunnels: • IPv4 packet in an IPv6 tunnel. • IPv6 packet in an IPv6 tunnel.	 No No	 No No
Packet is an IPv4 fragment.	Yes	No[2]
Packet is greater than 1518, 1522 or 1526 bytes (LPE=1b)[3].	Yes	Yes
Packet has 802.3ac tag.	Yes	Yes
IPv4 packet has IP options (IP header is longer than 20 bytes).	Yes	Yes
Packet has TCP or UDP options.	Yes	Yes
IP header's protocol field contains a protocol number other than TCP or UDP.	Yes	No

1. The IPv6 header portion can include supported extension headers as described in the IPv6 filter section.
2. UDP checksum of first fragment is supported.
3. Depends on number of VLAN tags.

26.1.11.1 Filters Details

The previous table lists general details about what packets are processed. In more detail, the packets are passed through a series of filters to determine if a receive checksum is calculated:

26.1.11.1.1 MAC Address Filter

This filter checks the MAC destination address to be sure it is valid (such as IA match, broadcast, multicast, etc.). The receive configuration settings determine which MAC addresses are accepted. See the various receive control configuration registers such as RCTL (RCTL.UPE, RCTL.MPE, RCTL.BAM), MTA, RAL, and RAH.

26.1.11.1.2 SNAP/VLAN Filter

This filter checks the next headers looking for an IP header. It is capable of decoding Ethernet II, Ethernet SNAP, and IEEE 802.3ac headers. It skips past any of these intermediate headers and looks for the IP header. The receive configuration settings determine which next headers are accepted. See the various receive control configuration registers such as RCTL (RCTL.VFE), VET, and VFTA.

26.1.11.1.3 IPv4 Filter

This filter checks for valid IPv4 headers. The version field is checked for a correct value (4).

IPv4 headers are accepted if they are any size greater than or equal to five (Dwords). If the IPv4 header is properly decoded, the IP checksum is checked for validity. The *RXCSUM.IPOFL* bit must be set for this filter to pass.

26.1.11.1.4 IPv6 Filter

This filter checks for valid IPpv6 headers, which are a fixed size and have no checksum. The IPv6 extension headers accepted are: hop-by-hop, destination options, and routing. The maximum size next header accepted is 16 Dwords (64 bytes).

26.1.11.1.5 IPv6 Extension Headers

IPv4 and TCP provide header lengths, which enable hardware to easily navigate through these headers on packet reception for calculating checksum and CRCs, etc. For receiving IPv6 packets; however, there is no IP header length to help hardware find the packet's ULP (such as TCP or UDP) header. One or more IPv6 extension headers might exist in a packet between the basic IPv6 header and the ULP header. The hardware must skip over these extension headers to calculate the TCP or UDP checksum for received packets.

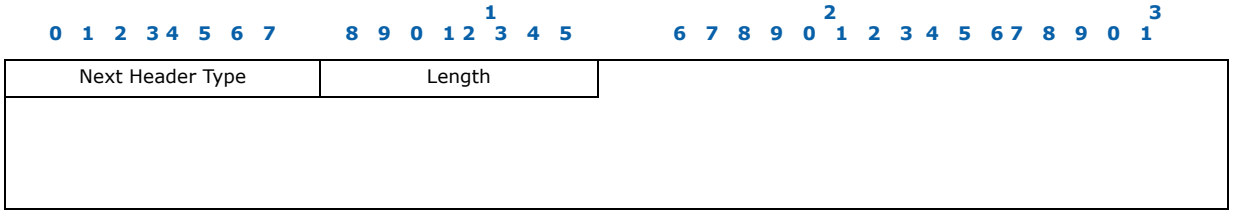
The IPv6 header length without extensions is 40 bytes. The IPv6 field *Next Header Type* indicates what type of header follows the IPv6 header at offset 40. It might be an upper layer protocol header such as TCP or UDP (*Next Header Type* of 6 or 17, respectively), or it might indicate that an extension header follows. The final extension header indicates with its *Next Header Type* field the type of ULP header for the packet.

IPv6 extension headers have a specified order. However, destinations must be able to process these headers in any order. Also, IPv6 (or IPv4) might be tunneled using IPv6, and thus another IPv6 (or IPv4) header and potentially its extension headers might be found after the extension headers.

The IPv4 *Next Header Type* is at byte offset nine. In IPv6, the first *Next Header Type* is at byte offset six.

All IPv6 extension headers have the *Next Header Type* in their first eight bits. Most have the length in the second eight bits (Offset Byte[1]) as shown:

Table 26-19. Typical IPv6 Extended Header Format (Traditional Representation)



The following table lists the encoding of the *Next Header Type* field and information on determining each header type's length. The IPv6 extension headers are not otherwise processed by the Controller so their details are not covered here.

Table 26-20. Header Type Encoding and Lengths

Header	Next Header Type	Header Length (Units are Bytes Unless Otherwise Specified)
IPv6	6	Always 40 bytes
IPv4	4	Offset Bits[7:4] Unit = 4 bytes
TCP	6	Offset Byte[12].Bits[7:4] Unit = 4 bytes
UDP	17	Always 8 bytes
Hop by Hop Options	0 (Note 1)	8+Offset Byte[1]
Destination Options	60	8+Offset Byte[1]
Routing	43	8+Offset Byte[1]
Fragment	44	Always 8 bytes
Authentication	51	8+4*(Offset Byte[1])
Encapsulating Security Payload	50	Note 3
No Next Header	59	Note 2

Notes:
1. Hop-by-hop options header is only found in the first *Next Header Type* of an IPv6 header.
2. When a *No Next Header* type is encountered, the rest of the packet should not be processed.
3. Encapsulated security payload - the Controller cannot offload packets with this header type.

The Controller hardware acceleration does not support all IPv6 extension header types (see Table 26-18).

The *RFCTL.Ipv6_DIS* bit must be cleared for this filter to pass.

26.1.11.1.6 UDP/TCP Filter

This filter checks for a valid UDP or TCP header. The prototype next header values are 0x11 and 0x06, respectively. The *RXCSUM.TUOFL* bit must be set for this filter to pass.

26.1.11.2 Receive UDP Fragmentation Checksum

The Controller might provide receive fragmented UDP checksum offload. The Controller should be configured in the following manner to enable this mode:

The *RXCSUM.PCSD* bit should be cleared. The *Packet Checksu*m and *IP Identification* fields are mutually exclusive with the RSS hash. When the RXCSUM.*PCSD* bit is cleared, *Packet Checksum* and *IP Identification* are active instead of RSS hash.

The *RXCSUM.IPPCSE* bit should be set. This field enables the IP payload checksum enable that is designed for the fragmented UDP checksum.

The *RXCSUM.PCSS* field must be zero. The packet checksum start should be zero to enable auto-start of the checksum calculation. The following table lists the exact description of the checksum calculation.

The following table also lists the outcome descriptor fields for the following incoming packets types:

Table 26-21. Descriptor Fields

Incoming Packet Type	Fragment Checksum	UDPV	UDPCS / L4CS
Non IP Packet	0b	0b	0b / 0b
Ipv6 Packet	0b	0b	Depends on transport header.
Non fragmented Ipv4 packet	0b	0b	Depends on transport header.
Fragmented Ipv4, when not first fragment	The unadjusted one's complement checksum of the IP payload.	0b	1b / 0b
Fragmented Ipv4, for the first fragment	Same as above	1 if the UDP header checksum is valid (not zero)	1b / 0b

Note: When the software device driver computes the 16-bit ones complement, the sum on the incoming packets of the UDP fragments, it should expect a value of 0xFFFF. See Section 26.1.11 for supported packet formats.

26.1.12 SCTP Offload

If a receive packet is identified as SCTP, the Controller checks the CRC32 checksum of this packet and identifies this packet as SCTP. Software is notified of the CRC check via the *L4I* bit in the *Extended Status* field of the Rx descriptor. The detection of a SCTP packet is indicated via the *SCTP* bit in the *packet Type* field of the Rx descriptor. The checker assumes the following SCTP packet format:

Table 26-22. SCTP Header

0 1 2 3 4 5 6 7	1 8 9 0 1 2 3 4 5	2 6 7 8 9 0 1 2 3	3 4 5 6 7 8 9 0 1
Source Port		Destination Port	
Verification Tag			
Checksum			
Chunks 1...n			

26.2 Transmit Functionality

26.2.1 Packet Transmission

Output packets are made up of pointer-length pairs constituting a descriptor chain (descriptor based transmission). Software forms transmit packets by assembling the list of pointer-length pairs, storing this information in the transmit descriptor, and then updating the on-chip transmit tail pointer to the descriptor. The transmit descriptor and buffers are stored in host memory. Hardware typically transmits the packet only after it has completely fetched all the L2 packet data from host memory and deposited it into the on-chip transmit FIFO. This permits TCP or UDP checksum computation and avoids problems with PCIe* under-runs.

Another transmit feature of the Controller is TCP/UDP segmentation. The hardware has the capability to perform packet segmentation on large data buffers offloaded from the Network Operating System (NOS). This feature is discussed in detail in Section 26.2.4.

In addition, the Controller supports SCTP offloading for transmit requests. See section Section 26.2.5.3 for details about SCTP.

26.2.1.1 Transmit Data Storage

Data is stored in buffers pointed to by the descriptors. Alignment of data is on an arbitrary byte boundary with the maximum size per descriptor limited only to the maximum allowed packet size (9728 bytes). A packet typically consists of two (or more) buffers, one (or more) for the header and one for the actual data. Each buffer is referenced by a different descriptor. Some software implementations copy the header(s) and packet data into one buffer and use only one descriptor per transmitted packet.

26.2.1.2 On-Chip Transmit Buffers

The Controller allocates by default a 20KB on-chip packet buffer per port. The buffer can be used to store packets until they are transmitted on the line. The Controller utilizes a single common ram structure for the on-chip transmit buffers allocated to the various ports. If a port is disabled, so that it can't be accessed by host and management, by setting EEPROM bit *LAN_DIS* in the "*Software Defined Pins Control*" word for the relevant port to 1.

The freed buffer space can be allocated to the active ports via the "*Initialization Control 4*" EEPROM word. Actual on-chip transmit buffer allocated to the port can be read in the *ITPBS register.*

26.2.1.3 On-Chip Descriptor Buffers

The Controller contains a 24 descriptor cache for each transmit queue used to reduce the latency of packet processing and to optimize the usage of the PCIe* bandwidth by fetching and writing back descriptors in bursts. The fetch and writeback algorithm are described in Section 26.2.2.5 and Section 26.2.2.6.

26.2.1.4 Transmit Contexts

The Controller provides hardware checksum offload and TCP/UDP segmentation facilities. These features enable TCP and UDP packet types to be handled more efficiently by performing additional work in hardware, thus reducing the software overhead associated with preparing these packets for transmission. Part of the parameters used by these features is handled though contexts.

A context refers to a set of device registers loaded or accessed as a group to provide a particular function. The Controller supports 2x8 context register sets (two per queue) per port on-chip. The transmit queues can contain transmit data descriptors (similar to the receive queue) as well as transmit context descriptors.

The contexts are queue specific and one context cannot be reused from one queue to another. This differs from the method used in previous devices that supported a pool of contexts to be shared between queues.

A transmit context descriptor differs from a data descriptor as it does not point to packet data. Instead, this descriptor provides the ability to write to the on-chip context register sets that support the transmit checksum offloading and the segmentation features of the Controller.

The Controller supports one type of transmit context. This on-chip context is written with a transmit context descriptor DTYP=2 and is always used as context for transmit data descriptor DTYP=3.

The *IDX* field contains an index to one of the two queue contexts. Software must track what context is stored in each *IDX* location.

Each advanced data descriptor that uses any of the advanced offloading features must refer to a context.

Contexts can be initialized with a transmit context descriptor and then used for a series of related transmit data descriptors. The context, for example, defines the checksum and offload capabilities for a given type of TCP/IP flow. All packets of this type can be sent using this context.

Software is responsible for ensuring that a context is only overwritten when it is no longer needed. Hardware does not include any logic to manage the on-chip contexts; it is completely up to software to populate and then use the on-chip context table.

Note: Software should not queue more than 2 context descriptors in sequence without an intervening data descriptor, to achieve adequate performance.

Each context defines information about the packet sent including the total size of the MAC header (*TDESC.MACHDR*), the maximum amount of payload data that should be included in each packet (*TDESC.MSS*), TCP header length (*TDES.TCPHDR*), IP header length (*TDESC.IPHDR*), and information about what type of protocol (TCP, IP, etc.) is used. Other than TCP, IP (*TDESC.TUCMD*), most information is specific to the segmentation capability.

Because there are dedicated on-chip resources for contexts, they remain constant until they are modified by another context descriptor. This means that a context can be used for multiple packets (or multiple segmentation blocks) unless a new context is loaded prior to each new packet. Depending on the environment, it might be unnecessary to load a new context for each packet. For example, if most traffic generated from a given node is standard TCP frames, this context could be setup once and used for many frames. Only when some other frame type is required would a new context need to be loaded by software. This new context could use a different index or the same index.

This same logic can also be applied to the TCP/UDP segmentation scenario, though the environment is a more restrictive one. In this scenario, the host is commonly asked to send messages of the same type, TCP/IP for instance, and these messages also have the same Maximum Segment Size (MSS). In this instance, the same context could be used for multiple TCP messages that require hardware segmentation.

26.2.2 Transmit Descriptors

The PCH supports legacy descriptors and advanced descriptors.

Legacy descriptors are intended to support legacy drivers to enable fast platform power up and to facilitate debug.

These descriptors should not be used when advanced features such as virtualization are used. The Legacy descriptors are recognized as such based on the *DEXT* bit as discussed later in this section.

In addition, the Controller supports two types of advanced transmit descriptors:

1. Advanced Transmit Context Descriptor, DTYP = 0010b.
2. Advanced Transmit Data Descriptor, DTYP = 0011b.

Note: DTYP values 0000b and 0001b are reserved.

The transmit data descriptor (both legacy and advanced) points to a block of packet data to be transmitted. The advanced transmit context descriptor does not point to packet data. It contains control/context information that is loaded into on-chip registers that affect the processing of packets for transmission. The following sections describe the descriptor formats.

26.2.2.1 Legacy Transmit Descriptor Format

Legacy descriptors are identified by having bit 29 of the descriptor (*TDESC.DEXT*) set to 0b. In this case, the descriptor format is defined as shown in Table 26-23. The address and length must be supplied by software. Also, bits in the command byte are optional, as are the CSO, and CSS fields.

Table 26-23. Transmit Descriptor (TDESC) Fetch Layout - Legacy Mode

	63 48	47 40	39 36	35 32	31 24	23 16	15 0
0	Buffer Address [63:0]						
8	VLAN	CSS	ExtCMD	STA	CMD	CSO	Length

Table 26-24. Transmit Descriptor (TDESC) Write-Back Layout - Legacy Mode

	63 48	47 40	39 36	35 32	31 24	23 16	15 0
0	Reserved				Reserved		
8	VLAN	CSS	Reserved	STA	CMD	CSO	Length

Note: For frames that span multiple descriptors, the VLAN, CSS, CSO, CMD.VLE, CMD.IC, and CMD.IFCS are valid only in the first descriptors and are ignored in the subsequent ones.

26.2.2.1.1 Length

Length (*TDESC.LENGTH*) specifies the length in bytes to be fetched from the buffer address provided.

The maximum length associated with any single legacy descriptor is 9728 bytes.

Descriptor length(s) might be limited by the size of the transmit FIFO. All buffers comprising a single packet must be able to be stored simultaneously in the transmit FIFO. For any individual packet, the sum of the individual descriptors' lengths must be below 9728 bytes.

Note: The maximum allowable packet size for transmits can change, based on the value written to the *DMA TX Max Allowable packet size* (*DTXMXPKTSZ*) register.

- Descriptors with zero length (null descriptors) transfer no data. Null descriptors can only appear between packets and must have their *EOP* bits set.
- If the *TCTL.PSP* bit is set, the total length of the packet transmitted, not including FCS should be at least 17 bytes.

26.2.2.1.2 Checksum Offset and Start - CSO and CSS

A *Checksum Offset* (*TDESC.CSO*) field indicates where, relative to the start of the packet, to insert a TCP checksum if this mode is enabled. A *Checksum Start* (*TDESC.CSS*) field indicates where to begin computing the checksum.

Both CSO and CSS are in units of bytes and must be in the range of data provided in the descriptors. For short packets that are not padded by software, CSS and CSO must be in the range of the unpadded data length, not the eventual padded length (64 bytes).

CSO must be set to the location of TCP checksum in the packet. CSS must be set to the beginning of the IP header or the L4 (TCP) header. Checksum calculation is not done if CSO or CSS are out of range. This occurs if (CSS > length) OR (CSO > length - 1).

In the case of an 802.1Q header, the offset values depend on the VLAN insertion enable (*VLE*) bit. If they are not set (VLAN tagging included in the packet buffers), the offset values should include the VLAN tagging. If these bits are set (VLAN tagging is taken from the packet descriptor), the offset values should exclude the VLAN tagging.

Note: Software must compute an offsetting entry to back out the bytes of the header that are not part of the IP pseudo header and should not be included in the TCP checksum and store it in the position where the hardware computed checksum is to be inserted. Hardware does not add the 802.1Q Ethertype or the VLAN field following the 802.1Q Ethertype to the checksum. So for VLAN packets, software can compute the values to back out only on the encapsulated packet rather than on the added fields.

- UDP checksum calculation is not supported by the legacy descriptors. When using legacy descriptors, the Controller is not aware of the L4 type of the packet and thus, does not support the translation of a checksum result of 0x0000 to 0xFFFF needed to differentiate between an UDP packet with a checksum of zero and an UDP packet without checksum.

Because the *CSO* field is eight bits wide, it puts a limit on the location of the checksum to 255 bytes from the beginning of the packet.

Hardware adds the checksum to the field at the offset indicated by the *CSO* field. Checksum calculations are for the entire packet starting at the byte indicated by the *CSS* field. A value of zero corresponds to the first byte in the packet.

CSS must be set in the first descriptor for a packet.

Table 26-25. Transmit Command (TDESC.CMD) Layout

7	6	5	4	3	2	1	0
RSV	VLE	DEXT	Rsv	RS	IC	IFCS	EOP

26.2.2.1.3 Command Byte - CMD

The CMD byte stores the applicable command and has the fields shown in Figure 26-25.

- RSV (bit 7) - Reserved
- VLE (bit 6) - VLAN Packet Enable
- DEXT (bit 5) - Descriptor Extension (0 for legacy mode)
- Reserved (bit 4) - Reserved
- RS (bit 3) - Report Status
- IC (bit 2) - Insert Checksum
- IFCS (bit 1) - Insert FCS
- EOP (bit 0) - End of Packet

VLE: Indicates that the packet is a VLAN packet. For example, hardware should add the VLAN Ethertype and an 802.1q VLAN tag to the packet.

Table 26-26. VLAN Tag Insertion Decision Table

VLE	Action
0b	Send generic Ethernet packet unless relevant bit in VMVIR registers is set.
1b	Send 802.1Q packet; the Ethernet *Type* field comes from the VET register and the VLAN data comes from the *VLAN* field of the TX descriptor.

RS: Signals the hardware to report the status information. This is used by software that does in-memory checks of the transmit descriptors to determine which ones are done. For example, if software queues up 10 packets to transmit, it can set the *RS* bit in the last descriptor of the last packet. If software maintains a list of descriptors with the *RS* bit set, it can look at them to determine if all packets up to (and including) the one with the *RS* bit set have been buffered in the output FIFO. Looking at the status byte and checking the *Descriptor Done* (*DD*) bit enables this operation. If *DD* is set, the descriptor has been processed. See Table 26-27 for the layout of the status field.

IC: If set, requests hardware to add the checksum of the data from *CSS* to the end of the packet at the offset indicated by the *CSO* field.

IFCS: When set, hardware appends the MAC FCS at the end of the packet. When cleared, software should calculate the FCS for proper CRC check. There are several cases in which software must set IFCS:

- Transmitting a short packet while padding is enabled by the *TCTL.PSP* bit.
- Checksum offload is enabled by the *IC* bit in the TDESC.CMD.
- VLAN header insertion enabled by the *VLE* bit in the TDESC.CMD or by the VMVIR registers.

EOP: When set, indicates this is the last descriptor making up the packet. More than one descriptor can be used to form a packet.

Note: As opposed to , VLE, IFCS, CSO, and IC must be set correctly only in the first descriptor of each packet. In previous silicon generations, some of these bits were required to be set in the last descriptor of a packet.

26.2.2.1.4 Status – STA

Table 26-27. Transmit Status (TDESC.STA) Layout

3	2	1	0
Reserved			DD

26.2.2.1.5 DD (Bit 0) - Descriptor Done Status

The DD bit provides the transmit status, when *RS* is set in the command: *DD* indicates that the descriptor is done and is written back after the descriptor has been processed.

Note: When head write back is enabled (*TDWBAL[n].Head_WB_En* = 1), the write-back of the *DD* bit to the descriptor is not executed.

26.2.2.1.6 VLAN

The *VLAN* field is used to provide the 802.1q/802.1ac tagging information. The *VLAN* field is valid only on the first descriptor of each packet when the *VLE* bit is set. The rule for VLAN tag is to use network ordering. The VLAN field is placed in the transmit descriptor in the following manner:

Table 26-28. VLAN Field (TDESC.VLAN) Layout

15 13	12	11 0
PRI	CFI	VLAN ID

- VLAN ID - the 12-bit tag indicating the VLAN group of the packet.

- Canonical Form Indication (CFI) - Set to zero for Ethernet packets.
- PRI - indicates the priority of the packet.

Note: The VLAN tag should be sent in network order (also called big endian).

26.2.2.2 Advanced Transmit Context Descriptor

Table 26-29. Transmit Context Descriptor (TDESC) Layout - (Type = 0010b)

	63 40	39 32	31 16	15 9	8 0
0	Reserved	Reserved	VLAN	MACLEN	IPLEN

	63 48	47 40	39	38 36	35 30	29	28 24	23 20	19 9	8 0
8	MSS	L4LEN	RSV[1]	IDX		DEXT	RSV[1]	DTYP	TUCMD	Reserved

1. RSV - Reserved

26.2.2.2.1 IPLEN (9)

IP header length. If an offload is requested, IPLEN must be greater than or equal to 20 and less than or equal to 511.

26.2.2.2.2 MACLEN (7)

This field indicates the length of the MAC header. When an offload is requested (either TSE or IXSM or TXSM is set), MACHDR must be larger than or equal to 14 and less than or equal to 127. This field should include only the part of the L2 header supplied by the software device driver and not the parts added by hardware. The following table lists the value of MACLEN in the different cases.

Table 26-30. MACLEN Values

SNAP	Regular VLAN	Extended VLAN	MACLEN
No	By hardware or no VLAN	No	14
No	By hardware or no VLAN	Yes	18
No	By software	No	18
No	By software	Yes	22
Yes	By hardware or no VLAN	No	22
Yes	By hardware or no VLAN	Yes	26
Yes	By software	No	26
Yes	By software	Yes	30

VLAN (16) - 802.1Q VLAN tag to be inserted in the packet during transmission. This VLAN tag is inserted and needed only when a packet using this context has its *DCMD.VLE* bit set. This field should include the entire 16-bit *VLAN* field including the *CFI* and *Priority* fields as shown in Table 26-28.

Note: The VLAN tag should be sent in network order.

26.2.2.2.3 TUCMD (11)

Table 26-31. Transmit Command (TDESC.TUCMD) Layout

10 6	5	4	3 2	1	0
Reserved	Reserved	Reserved	L4T	IPV4	SNAP

- RSV (bit 10:6) - Reserved
- RSV (bit 5:4) - Reserved
- L4T (bit 3:2) - L4 Packet TYPE (00b: UDP; 01b: TCP; 10b: SCTP; 11b: Reserved)
- IPV4 (bit 1) - IP Packet Type: When 1b, Ipv4; when 0b, Ipv6
- SNAP (bit 0) - SNAP indication

26.2.2.2.4 DTYP(4)

Always 0010b for this type of descriptor.

26.2.2.2.5 DEXT(1)

Descriptor Extension (1b for advanced mode).

26.2.2.2.6 IDX (3)

Index into the hardware context table where this context is stored. In the Controller, the 2 available register context sets per queue are accessed using the LSB bit and the two MSB bits are reserved and should always be 0.

26.2.2.2.7 L4LEN (8)

Layer 4 header length. If *TSE* is set in the data descriptor pointing to this context, this field must be greater than or equal to 12 and less than or equal to 255. Otherwise, this field is ignored.

26.2.2.2.8 MSS (16)

Controls the Maximum Segment Size (MSS). This specifies the maximum TCP payload segment sent per frame, not including any header or trailer. The total length of each frame (or section) sent by the TCP/UDP segmentation mechanism (excluding Ethernet CRC) as follows:

Total length is equal to:

MACLEN + 4(if VLE set) + 4 or 8(if CMTGI is set or if also RLTTGI is set - assuming BCNTLEN is clear) + IPLEN + L4LEN + MSS

The one exception is the last packet of a TCP/UDP segmentation, which is typically shorter.

MSS is ignored when *DCMD.TSE* is not set.

Note: The headers lengths must meet the following:

MACLEN + IPLEN + L4LEN <= 512

- The MSS value should be larger than 0 and the maximum MSS value should not exceed 9216 bytes (9KB) length.

The context descriptor requires valid data only in the fields used by the specific offload options. The following table lists the required valid fields according to the different offload options.

Table 26-32. Valid Field in Context vs. Required Offload

Required Offload			Valid Fields in Context						
TSE	**TXSM**	**IXSM**	**VLAN**	**L4LEN**	**IPLEN**	**MACLEN**	**MSS**	**L4T**	**IPV4**
1b[1]	1b	X[2]	VLE	Yes	Yes	Yes	Yes	Yes	Yes
0b	1b	X[2]	VLE	No	Yes	Yes	No	Yes	Yes
0b	0b	1b	VLE	No	Yes	Yes	No	No	Yes
0b	0b	0b	No context required unless VLE is set.						

1. If TSE is set, TXSM must be set to 1.
2. X - don't care

26.2.2.3 Advanced Transmit Data Descriptor

Table 26-33. Advanced Transmit Data Descriptor (TDESD) Layout - (Type = 0011b)

0	Address[63:0]									
8	PAYLEN	POPTS		IDX	STA	DCMD	DTYP	MAC	RSV[1]	DTALEN
	63 46	**45 40**	**39**	**38 36**	**35 32**	**31 24**	**23 20**	**19 18**	**17 16**	**15 0**

1. RSV - Reserved

Table 26-34. Advanced Tx Descriptor Write-back Format

0	RSV[1]		
8	Reserved	STA	Reserved
	63 36	**35 32**	**31 0**

1. RSV - Reserved

Note: For frames that span multiple descriptors, all fields **apart** from DCMD.EOP, DCMD.RS, DCMD.DEXT, DTALEN, Address and DTYP are valid only in the first descriptor and are ignored in the subsequent ones.

26.2.2.3.1 Address (64)

Physical address of a data buffer in host memory that contains a portion of a transmit packet.

26.2.2.3.2 DTALEN (16)

Length in bytes of data buffer at the address pointed to by this specific descriptor.

Note: If the *TCTL.PSP* bit is set, the total length of the packet transmitted, not including FCS, should be at least 17 bytes.

- The maximum allowable packet size for transmits is based on the value written to the *DMA TX Max Allowable packet size* (*DTXMXPKTSZ*) register. Default value is 9728 bytes.

26.2.2.3.3 MAC (2)

Table 26-35. Transmit Data (TDESD.MAC) Layout

bit 1	bit 0
1588	Reserved

- Reserved (bit 0)
- 1588 (bit 1) - IEEE1588 Timestamp packet.

26.2.2.3.4 DTYP (4)

0011b is the value for this descriptor type.

26.2.2.3.5 DCMD (8)

Table 26-36. Transmit Data (TDESD.DCMD) Layout

7	6	5	4	3	2	1	0
TSE	VLE	DEXT	Reserved	RS	Reserved	IFCS	EOP

- TSE (bit 7) - TCP/UDP Segmentation Enable
- VLE (bit 6) - VLAN Packet Enable
- DEXT (bit 5) - Descriptor Extension (1b for advanced mode)
- Reserved (bit 4)
- RS (bit 3) - Report Status
- Reserved (bit 2)
- IFCS (bit 1) - Insert FCS
- EOP (bit 0) - End Of Packet

TSE indicates a TCP/UDP segmentation request. When *TSE* is set in the first descriptor of a TCP packet, hardware must use the corresponding context descriptor in order to perform TCP segmentation. The type of segmentation applied is defined according to the *TUCMD.L4T* field in the context descriptor.

Note: It is recommended that *TCTL.PSP* be enabled when *TSE* is used since the last frame can be shorter than 60 bytes - resulting in a bad frame if TCTL.*PSP* is disabled.

VLE indicates that the packet is a VLAN packet and hardware must add the VLAN Ethertype and an 802.1q VLAN tag to the packet.

DEXT must be 1b to indicate advanced descriptor format (as opposed to legacy).

RS signals hardware to report the status information. This is used by software that does in-memory checks of the transmit descriptors to determine which ones are done. For example, if software queues up 10 packets to transmit, it can set the *RS* bit in the last descriptor of the last packet. If software maintains a list of descriptors with the *RS* bit set, it can look at them to determine if all packets up to (and including) the one with

the *RS* bit set have been buffered in the output FIFO. Looking at the status byte and checking the *DD* bit do this. If *DD* is set, the descriptor has been processed. See the sections that follow for the layout of the status field.

Note: Descriptors with zero length transfer no data.

IFCS, when set, hardware appends the MAC FCS at the end of the packet. When cleared, software should calculate the FCS for proper CRC check. There are several cases in which software must set IFCS:

- Transmitting a short packet while padding is enabled by the *TCTL.PSP* bit.
- Checksum offload is enabled by the either the *TXSM* or *IXSM* bits in the TDESD.POPTS field.
- VLAN header insertion enabled by the *VLE* bit in the TDESD.DCMD.
- TCP/UDP segmentation offload enabled by *TSE* bit in the TDESD.DCMD.

EOP indicates whether this is the last buffer for an incoming packet.

26.2.2.3.6 STA (4)

- Rsv (bits 1-3) - Reserved
- DD (bit 0) - Descriptor Done

26.2.2.3.7 IDX (3)

Index into the hardware context table to indicate which context should be used for this request. If no offload is required, this field is not relevant and no context needs to be initiated before the packet is sent. See Table 26-32 for details in which packets require a context reference.

26.2.2.3.8 POPTS (6)

Table 26-37. Transmit Data (TDESD.POPTS) Layout

5 3	2	1	0
Reserved	Reserved	TXSM	IXSM

- Reserved (bits 5:3)
- Reserved (bit 2)
- TXSM (bit 1) - Insert L4 Checksum
- IXSM (bit 0) - Insert IP Checksum

TXSM, when set to 1b, L4 checksum must be inserted. In this case, TUCMD.L4T in the context descriptor indicates whether the checksum is TCP, UDP, or SCTP.

When *DCMD.TSE* in TDESD is set, *TXSM* must be set to 1b.

If this bit is set, the packet should at least contain a TCP header.

IXSM, when set to 1b, indicates that IP checksum must be inserted. For IPv6 packets this bit must be cleared.

If the *DCMD.TSE* bit is set in data descriptor, and *TUCMD.IPV4* is set in context descriptor, POPTS.*IXSM* must be set to 1b as well.

If this bit is set, the packet should at least contain an IP header.

26.2.2.3.9 PAYLEN (18)

PAYLEN indicates the size (in byte units) of the data buffer(s) in host memory for transmission. In a single send packet, PAYLEN defines the entire packet size fetched from host memory. It does not include the fields that hardware adds such as: optional VLAN tagging, Ethernet CRC or Ethernet padding. In a large send case (regardless if it is transmitted on a single or multiple packets), PAYLEN defines the protocol payload size fetched from host memory. In TCP or UDP segmentation offload, PAYLEN defines the TCP/UDP payload size.

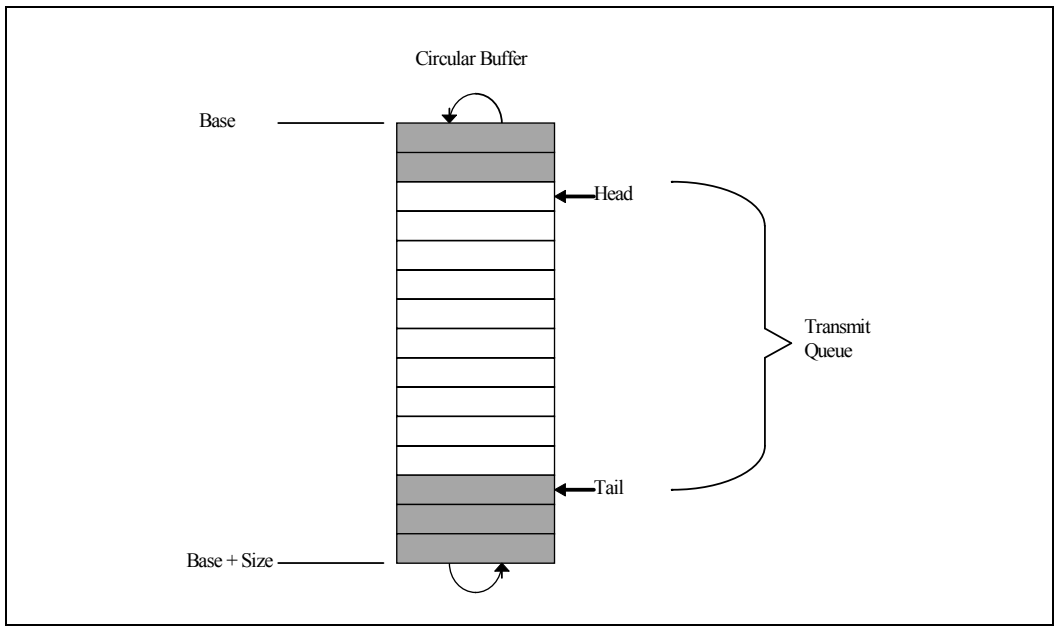
Note: When a packet spreads over multiple descriptors, all the descriptor fields are only valid in the first descriptor of the packet, except for *RS*, which is always checked, and EOP, which is always set at last descriptor of the series.

26.2.2.4 Transmit Descriptor Ring Structure

The transmit descriptor ring structure is shown in Figure 26-19. A pair of hardware registers maintains each transmit descriptor ring in the host memory. New descriptors are added to the queue by software by writing descriptors into the circular buffer memory region and moving the tail pointer associated with that queue. The tail pointer points to one entry beyond the last hardware owned descriptor. Transmission continues up to the descriptor where head equals tail at which point the queue is empty.

Descriptors passed to hardware should not be manipulated by software until the head pointer has advanced past them.

Figure 26-19.Transmit Descriptor Ring Structure



The shaded boxes in the figure represent descriptors that are not currently owned by hardware that software can modify.

The transmit descriptor ring is described by the following registers:

- Transmit Descriptor Base Address register (*TDBA 0-7*):

This register indicates the start address of the descriptor ring buffer in the host memory; this 64-bit address is aligned on a 16-byte boundary and is stored in two consecutive 32-bit registers. Hardware ignores the lower four bits.

- Transmit Descriptor Length register (*TDLEN 0-7*):
  This register determines the number of bytes allocated to the circular buffer. This value must be zero modulo 128.
- Transmit Descriptor Head register (*TDH 0-7*):
  This register holds a value that is an offset from the base and indicates the in-progress descriptor. There can be up to 64 KB descriptors in the circular buffer. Reading this register returns the value of head corresponding to descriptors already loaded in the output FIFO. This register reflects the internal head of the hardware write-back process including the descriptor in the posted write pipe and might point further ahead than the last descriptor actually written back to the memory.
- Transmit Descriptor Tail register (*TDT 0-7*):
  This register holds a value, which is an offset from the base, and indicates the location beyond the last descriptor hardware can process. This is the location where software writes the first new descriptor.
  The driver should not handle to the Controller descriptors that describe a partial packet. Consequently, the number of descriptors used to describe a packet can not be larger than the ring size.

The base register indicates the start of the circular descriptor queue and the length register indicates the maximum size of the descriptor ring. The lower seven bits of length are hard wired to 0b. Byte addresses within the descriptor buffer are computed as follows: address = base + (ptr * 16), where ptr is the value in the hardware head or tail register.

The size chosen for the head and tail registers permit a maximum of 65528 (64 KB) descriptors, or approximately 16 KB packets for the transmit queue given an average of four descriptors per packet.

Once activated, hardware fetches the descriptor indicated by the hardware head register. The hardware tail register points one descriptor beyond the last valid descriptor. Software can read and detect which packets have already been processed by hardware as follows:

- Read the head register to determine which packets (those logically before the head) have been transferred to the on-chip FIFO or transmitted. This method is not recommended because races between the internal update of the head register and the actual write-back of descriptors might occur.
- Read the value of the head as stored at the address pointed by the *TDBAH/TDBAL* pair.
- Track the *DD* bits in the descriptor ring.

All the registers controlling the descriptor rings behavior should be set before transmit is enabled, apart from the tail registers which are used during the regular flow of data.

Note: Software can determine if a packet has been sent by either of three methods: setting the *RS* bit in the transmit descriptor command field or by performing a PIO read of the transmit head register, or by reading the head value written by the Controller to the address pointed by the *TDWBAL* and *TDWBAH* registers (see Section 26.2.3 for details). Checking the transmit descriptor *DD* bit or head value in memory eliminates a potential race condition. All descriptor data is written to the I/O bus prior to incrementing the head register, but a read of the head register could pass the data write in systems performing I/O write buffering. Updates to transmit descriptors use the same I/O write path and follow all data writes. Consequently, they are not subject to the race.

In general, hardware prefetches packet data prior to transmission. Hardware typically updates the value of the head pointer after storing data in the transmit FIFO.

26.2.2.5 Transmit Descriptor Fetching

The descriptor processing strategy for transmit descriptors is essentially the same as for receive descriptors except that a different set of thresholds are used. In addition the Controller enables assigning 2 priority levels to each transmit descriptor queue using the *TXDCTL[n].Priority* bit. A transmit descriptor belonging to a queue with the *TXDCTL[n].Priority* set to 1 will always be fetched prior to a transmit descriptor belonging to a queue with the *TXDCTL[n].Priority* bit reset to 0. The number of on-chip transmit descriptors per queue is 24.

When there is an on-chip descriptor buffer empty, a fetch happens as soon as any descriptors are made available (host writes to the tail pointer). If several on-chip descriptor queues are in this situation at the same time, the queue with the highest priority as defined by the *TXDCTL[n].Priority* bit, is fetched. If several on-chip transmit descriptor queues with the same priority need to fetch descriptors, descriptors from queues that are more starved are fetched. If a number of queues have a similar priority and starvation level, highest indexed queue is served first and so forth, down to the lowest indexed queue.

Note: The starvation level of a queue corresponds to the number of descriptors above the prefetch threshold (*TXDCTL[n].PTHRESH)* that are already in the internal queue. The queue is more starved if there are less descriptors in the internal transmit descriptor cache. Comparing starvation level might be done roughly, not at the single descriptor level of resolution.

A queue is considered empty for the transmit descriptor fetch algorithm as long as:

- There is still no complete packet (single or large send) in its corresponding internal queue.
- There is no descriptor already in its way from system memory to the internal cache.
- The internal corresponding internal descriptor cache is not full.

Each time a descriptor fetch request is sent for an empty queue, the maximum available number of descriptor is requested, regardless of cache alignment issues.

When the on-chip buffer is nearly empty (below *TXDCTL[n].PTHRESH*), a prefetch is performed each time enough valid descriptors (*TXDCTL[n].HTHRESH*) are available in host memory and no other DMA activity of greater priority is pending (descriptor fetches and write-backs or packet data transfers).

When the number of descriptors in host memory is greater than the available on-chip descriptor storage, the Controller might elect to perform a fetch that is not a multiple of cache-line size. Hardware performs this non-aligned fetch if doing so results in the next descriptor fetch being aligned on a cache-line boundary. This enables the descriptor fetch mechanism to be more efficient in the cases where it has fallen behind software.

Note: The Controller NEVER fetches descriptors beyond the descriptor tail pointer.

26.2.2.6 Transmit Descriptor Write-Back

The descriptor write-back policy for transmit descriptors is similar to that of the receive descriptors when the *TXDCTL[n].WTHRESH* value is not 0x0. In this case, all descriptors are written back regardless of the value of their *RS* bit.

When the *TXDCTL[n].WTHRESH* value is 0x0, since transmit descriptor write-backs do not happen for every descriptor, only transmit descriptors that have the *RS* bit set are written back.

Any descriptor write-back includes the full 16 bytes of the descriptor.

Since the benefit of delaying and then bursting transmit descriptor write-backs is small at best, it is likely that the threshold is left at the default value (0b) to force immediate write-back of transmit descriptors with their *RS* bit set and to preserve backward compatibility.

Descriptors are written back in one of three cases:

- *TXDCTL[n].WTHRESH* = 0x0 and a descriptor which has *RS* set is ready to be written back.
- The corresponding *EITR* counter has reached zero.
- *TXDCTL[n].WTHRESH* > 0x0 and *TXDCTL[n].WTHRESH* descriptors have accumulated.

For the first condition, write-backs are immediate. This is the default operation and is backward compatible with implementation.

The other two conditions are only valid if descriptor bursting is enabled (Section 28.11.1.15). In the second condition, the EITR counter is used to force timely write-back of descriptors. The first packet after timer initialization starts the timer. Timer expiration flushes any accumulated descriptors and sets an interrupt event (TXDW).

For the final condition, if *TXDCTL[n].WTHRESH* descriptors are ready for write-back, the write-back is performed.

An additional mode in which transmit descriptors are not written back at all and the head pointer of the descriptor ring is written instead as described in Section 26.2.3.

Note: When transmit ring is smaller than internal cache size (24 descriptors) then at list one full packet should be placed in the ring and *TXDCTL[n].WTHRESH* value should be less than ring size. If *TXDCTL[n].WTHRESH* is 0x0 (transmit RS mode) then at least one descriptor should have the RS ring set inside the ring.

26.2.3 Transmit Completions Head Write Back

In legacy hardware, transmit requests are completed by writing the *DD* bit to the transmit descriptor ring. This causes cache thrash since both the software device driver and hardware are writing to the descriptor ring in host memory. Instead of writing the *DD* bits to signal that a transmit request completed, hardware can write the contents of the descriptor queue head to host memory. The software device driver reads that memory location to determine which transmit requests are complete. In order to improve the performance of this feature, the software device driver needs to program DCA registers to configure which CPU is processing each TX queue.

26.2.3.1 Description

The head counter is reflected in a memory location that is allocated by software, for each queue.

Head write back occurs if *TDWBAL[n].Head_WB_En* is set for this queue and the *RS* bit is set in the Tx descriptor, following corresponding data upload into packet buffer. If the head write-back feature is enabled, the Controller ignores *WTRESH* and takes in

account only descriptors with the *RS* bit set (as if the *WTRESH* was set to 0b). In addition, the head write-back occurs upon EITR expiration for queues where the *WB_on_EITR* bit in *TDWBAL* is set.

Software can also enable coalescing of the head write-back operations to reduce traffic on the PCIe* bus, by programming the *TXDCTL.HWBTHRESH* field to a value greater than 0. In this case head write-back operation will occur only after the internal pending write-back count is greater than the *TXDCTL.HWBTHRESH* value.

The software device driver has control on this feature through Tx queue 0-7 head write-back address, low (*TDWBAL[n]*) and high *(TDWBAH[n]*) registers thus supporting 64-bit address access. See registers description in Section 28.11.1.16 and Section 28.11.1.17.

The 2 low register's LSB bits of the *TDWBAL[n]* register hold the control bits.

1. The *Head_WB_En* bit enables activation of the head write back feature. When *TDWBAL[n].Head_WB_En is set to 1* no TX descriptor write-back is executed for this queue.
2. *The WB_on_EITR* bit enables head write upon EITR expiration. When Head write back operation is enabled (*TDWBAL[n].Head_WB_En* = 1) setting the *TDWBAL[n].WB_on_EITR* bit to 1 enables placing an upper limit on delay of head write-back operation.

The 30 upper bits of the *TDWBAL[n]* register hold the lowest 32 bits of the head write-back address, assuming that the two last bits are zero. The *TDWBAH[n]* register holds the high part of the 64-bit address.

Note: Hardware writes a full Dword when writing this value, so software should reserve enough space for each head value.

- If software enables Head Write-Back, it must also disable PCI Express Relaxed Ordering on the write-back transactions. This is done by disabling bit 11 in the *TXCTL* register for each active transmit queue. See Section 28.12.1.2.
    - The Controller might update the Head with values that are larger then the last Head pointer which holds a descriptor with RS bit set, but still the value will always point to a free descriptor (free descriptor is one that is not owned by the Controller anymore).

26.2.4 TCP/UDP Segmentation

Hardware TCP segmentation is one of the offloading options supported by the Windows* and Linux* TCP/IP stack. This is often referred to as TCP Segmentation Offloading or TSO. This feature enables the TCP/IP stack to pass to the network device driver a message to be transmitted that is bigger than the Maximum Transmission Unit (MTU) of medium. It is then the responsibility of the software device driver and hardware to divide the TCP message into MTU size frames that have appropriate layer 2 (Ethernet), 3 (IP), and 4 (TCP) headers. These headers must include sequence number, checksum fields, options and flag values as required. Some of these values (such as the checksum values) are unique for each packet of the TCP message and other fields such as the source IP address are constant for all packets associated with the TCP message.

The Controller supports also UDP segmentation for embedded applications, although this offload is not supported by the regular Windows* and Linux* stacks. Any reference in this section to TCP segmentation, should be considered as referring to both TCP and UDP segmentation.

Padding (TCTL.PSP) must be enabled in TCP segmentation mode, since the last frame might be shorter than 60 bytes, resulting in a bad frame if *PSP* is disabled.

The offloading of these mechanisms from the software device driver to the Controller saves significant CPU cycles. The software device driver shares the additional tasks to support these options.

26.2.4.1 Assumptions

The following assumptions apply to the TCP segmentation implementation in the Controller:

- The *RS* bit operation is not changed.
- Interrupts are set after data in buffers pointed to by individual descriptors is transferred (DMA'd) to hardware.

26.2.4.2 Transmission Process

The transmission process for regular (non-TCP segmentation packets) involves:

- The protocol stack receives from an application a block of data that is to be transmitted.
- The protocol stack calculates the number of packets required to transmit this block based on the MTU size of the media and required packet headers.

For each packet of the data block:

- Ethernet, IP and TCP/UDP headers are prepared by the stack.
- The stack interfaces with the software device driver and commands it to send the individual packet.
- The software device driver gets the frame and interfaces with the hardware.
- The hardware reads the packet from host memory (via DMA transfers).
- The software device driver returns ownership of the packet to the Network Operating System (NOS) when hardware has completed the DMA transfer of the frame (indicated by an interrupt).

The transmission process for the Controller TCP segmentation offload implementation involves:

- The protocol stack receives from an application a block of data that is to be transmitted.
- The stack interfaces to the software device driver and passes the block down with the appropriate header information.
- The software device driver sets up the interface to the hardware (via descriptors) for the TCP segmentation context.

Hardware DMA's (transfers) the packet data and performs the Ethernet packet segmentation and transmission based on offset and payload length parameters in the TCP/IP context descriptor including:

- Packet encapsulation
- Header generation and field updates including IPv4, IPV6, and TCP/UDP checksum generation
- The software device driver returns ownership of the block of data to the NOS when hardware has completed the DMA transfer of the entire data block (indicated by an interrupt).

26.2.4.2.1 TCP Segmentation Data Fetch Control

To perform TCP Segmentation in the Controller, the DMA must be able to fit at least one packet of the segmented payload into available space in the on-chip Packet Buffer. The DMA does various comparisons between the remaining payload and the Packet Buffer available space, fetching additional payload and sending additional packets as space permits.

To support interleaving between descriptor queues at Ethernet frame resolution inside TSO requests, the frame header pointed to by the so called header descriptors are reread from system memory by hardware for every LSO segment. the Controller stores in an internal cache only the header's descriptors instead of the header's content.

To limit the internal cache size software should not spread the L3/L4 header (TCP, UDP, IPV4 or IPV6) on more than 4 descriptors. In the last header buffer it's allowed to mix header and data. This limitation stands for up to Layer4 header included, and for IPv4 or IPv6 indifferently.

26.2.4.2.2 TCP Segmentation Write-Back Modes

Since the TCP segmentation mode uses the buffers that contains the L3/L4 header multiple times, there are some limitations on the usage of different combinations of writeback and buffer release methods in order to guarantee the header buffer's availability until the entire packet is processed. These limitations are described in Table 26-38 below.

Table 26-38. Write Back Options For Large Send

WTHRESH	RS	HEAD Write Back Enable	Hardware Behavior	Software Expected Behavior for TSO packets.
0	Set in EOP descriptors only	Disable	Hardware writes back descriptors with RS bit set one at a time.	Software can retake ownership of all descriptors up to last descriptor with DD bit set.
0	Set in any descriptors	Disable	Hardware writes back descriptors with RS bit set one at a time.	Software can retake ownership of entire packets (EOP bit set) up to last descriptor with DD bit set.
0	Not set at all	Disable	Hardware does not write back any descriptor (since *RS* bit is not set)	Software should poll the TDH register. The TDH register reflects the last descriptor that software can take ownership of.[1]
0	Not set at all	Enable	Hardware writes back the head pointer only at EITR expire event reflecting the last descriptor that software can take ownership of.	Software may poll the TDH register or use the head value written back at EITR expire event. The TDH register reflects the last descriptor that software can take ownership of.
>0	Don't care	Disable	Hardware writes back all the descriptors in bursts and set all the *DD* bits.	Software can retake ownership of entire packets up to last descriptor with both DD and EOP bits set. Note: The TDH register reflects the last descriptor that software can take ownership of[1].
Don't care	Set in EOP descriptors only	Enable	Hardware writes back the Head pointer per each descriptor with *RS* bit set.[2]	Software can retake ownership of all descriptors up to the descriptor pointed by the head pointer read from system memory (by interrupt or polling).
Don't care	Set in any descriptors	Enable	Hardware writes back the Head pointer per each descriptor with *RS* bit set.	This mode is illegal since software won't access the descriptor, it cannot tell when the pointer passed the EOP descriptor.

1. Polling of the TDH register is a valid method only when the RS bit is never set, otherwise race conditions between software and hardware accesses to the descriptor ring can occur.
2. At EITR expire event, the Hardware writes back the head pointer reflecting the last descriptor that software can take ownership of.

26.2.4.3 TCP Segmentation Performance

Performance improvements for a hardware implementation of TCP Segmentation off-load include:

- The stack does not need to partition the block to fit the MTU size, saving CPU cycles.
- The stack only computes one Ethernet, IP, and TCP header per segment, saving CPU cycles.
- The Stack interfaces with the device driver only once per block transfer, instead of once per frame.
- Larger PCIe* bursts are used which improves bus efficiency (such as lowering transaction overhead).
- Interrupts are easily reduced to one per TCP message instead of one per packet.
- Fewer I/O accesses are required to command the hardware.

26.2.4.4 Packet Format

Typical TCP/IP transmit window size is 8760 bytes (about 6 full size frames). Today the average size on corporate Intranets is 12-14KB, and normally the maximum window size allowed is 64KB (unless Windows Scaling - RFC 1323 is specified). A TCP message can be as large as 256 KB and is generally fragmented across multiple pages in host memory. The Controller partitions the data packet into standard Ethernet frames prior to transmission. The Controller supports calculating the Ethernet, IP, TCP, and UDP headers, including checksum, on a frame-by-frame basis.

Table 26-39. TCP/IP or UDP/IP Packet Format Sent by Host

L2/L3/L4 Header			Data
Ethernet	IPv4/IPv6	TCP/UDP	DATA (full TCP message)

Table 26-40. TCP/IP or UDP/IP Packet Format Sent by the Controller

Pseudo Header (updated)	Data (first MSS)	FCS	...	Pseudo Header (updated)	Data (Next MSS)	FCS	...

Frame formats supported by the Controller include:

- Ethernet 802.3
- IEEE 802.1Q VLAN (Ethernet 802.3ac)
- Ethernet Type 2
- Ethernet SNAP
- IPv4 headers with options
- IPv6 headers with extensions
- TCP with options
- UDP with options.

VLAN tag insertion might be handled by hardware

Note: UDP (unlike TCP) is not a "reliable protocol", and fragmentation is not supported at the UDP level. UDP messages that are larger than the MTU size of the given network medium are normally fragmented at the IP layer. This is different from TCP, where large TCP messages can be fragmented at either the IP or TCP layers depending on the software implementation. The Controller has the ability to segment UDP traffic (in addition to TCP traffic), however, because UDP packets are generally fragmented at the IP layer, the Controller's "TCP Segmentation" feature is not normally useful to handle UDP traffic.

26.2.4.5 TCP/UDP Segmentation Indication

Software indicates a TCP/UDP Segmentation transmission context to the hardware by setting up a TCP/IP Context Transmit Descriptor (see Section 26.2.2). The purpose of this descriptor is to provide information to the hardware to be used during the TCP segmentation off-load process.

Setting the *TSE* bit in the *TDESD.DCMD* field to 1b indicates that this descriptor refers to the TCP Segmentation context (as opposed to the normal checksum off loading context). This causes the checksum off loading, packet length, header length, and maximum segment size parameters to be loaded from the Context descriptor into the device.

The TCP Segmentation prototype header is taken from the packet data itself. Software must identity the type of packet that is being sent (IPv4/IPv6, TCP/UDP, other), calculate appropriate checksum off loading values for the desired checksum, and calculate the length of the header which is pre-appended. The header might be up to 240 bytes in length.

Once the TCP Segmentation context has been set, the next descriptor provides the initial data to transfer. This first descriptor(s) must point to a packet of the type indicated. Furthermore, the data it points to might need to be modified by software as it serves as the prototype header for all packets within the TCP Segmentation context. The following sections describe the supported packet types and the various updates which are performed by hardware. This should be used as a guide to determine what must be modified in the original packet header to make it a suitable prototype header.

The following summarizes the fields considered by the driver for modification in constructing the prototype header.

IP Header

For IPv4 headers:

- *Identification* Field should be set as appropriate for first packet of send (if not already)
- Header Checksum should be zeroed out unless some adjustment is needed by the driver

TCP Header

- Sequence Number should be set as appropriate for first packet of send (if not already)
- PSH, and FIN flags should be set as appropriate for LAST packet of send
- TCP Checksum should be set to the partial pseudo-header sum as follows (there is a more detailed discussion of this is Section 26.2.4.6):

Table 26-41. TCP Partial Pseudo-Header Sum for IPv4

IP Source Address		
IP Destination Address		
Zero	Layer 4 Protocol ID	Zero

Table 26-42. TCP Partial Pseudo-Header Sum for IPv6

IPv6 Source Address	
IPv6 Final Destination Address	
Zero	
Zero	Next Header

UDP Header

- Checksum should be set as in TCP header, above

The following sections describe the updating process performed by the hardware for each frame sent using the TCP Segmentation capability.

26.2.4.6 Transmit Checksum Offloading with TCP/UD Segmentation

The Controller supports checksum off-loading as a component of the TCP Segmentation off-load feature and as a standalone capability. Section 26.2.5 describes the interface for controlling the checksum off-loading feature. This section describes the feature as it relates to TCP Segmentation.

The Controller supports IP and TCP header options in the checksum computation for packets that are derived from the TCP Segmentation feature.

Note: The Controller is capable of computing one level of IP header checksum and one TCP/UDP header and payload checksum. In case of multiple IP headers, the driver needs to compute all but one IP header checksum. The Controller calculates check sums on the fly on a frame-by-frame basis and inserts the result in the IP/TCP/UDP headers of each frame. TCP and UDP checksum are a result of performing the checksum on all bytes of the payload and the pseudo header.

Three specific types of checksum are supported by the hardware in the context of the TCP Segmentation off-load feature:

- IPv4 checksum
- TCP checksum

Each packet that is sent via the TCP segmentation off-load feature optionally includes the IPv4 checksum and either the TCP checksum.

All checksum calculations use a 16-bit wide one's complement checksum. The checksum word is calculated on the outgoing data.

Table 26-43. Supported Transmit Checksum Capabilities

Packet Type	Hardware IP Checksum Calculation	Hardware TCP/UDP Checksum Calculation
IP v4 packets	Yes	Yes
IP v6 packets (no I checksum in Ipv6)	NA	Yes

Table 26-43. Supported Transmit Checksum Capabilities

Packet is greater than 1518, 1522 or 1526 bytes (LPE=1b)[1]	Yes	Yes
Packet has 802.3ac tag	Yes	Yes
Packet has IP options (IP header is longer than 20 bytes)	Yes	Yes
Packet has TCP or UDP options	Yes	Yes
IP header's protocol field contains a protocol # other than TCP or UDP.	Yes	No

1. Depends on number of VLAN tags.

The table below summarizes the conditions of when checksum off loading can/should be calculated.

Table 26-44. Conditions for Checksum Off Loading

Packet Type	IPv4	TCP/UDP	Reason
Non TSO	Yes	No	IP Raw packet (non TCP/UDP protocol)
	Yes	Yes	TCP segment or UDP datagram with checksum off-load
	No	No	Non-IP packet or checksum not offloaded
TSO	Yes	Yes	For TSO, checksum off-load must be done

26.2.4.7 TCP/UDP/IP Header Update

IP/TCP or IP/UDP header is updated for each outgoing frame based on the IP/TCP header prototype which hardware DMA's from the first descriptor(s). The checksum fields and other header information are later updated on a frame-by-frame basis. The updating process is performed concurrently with the packet data fetch.

The following sections define what fields are modified by hardware during the TCP Segmentation process by the Controller.

Note: Software must make PAYLEN and HDRLEN value of Context descriptors correct. Otherwise, the failure of Large Send due to either under-run or over-run might cause hardware to send bad packets or even cause TX hardware to hang. The indication of Large Send failure can be checked in the TSCTFC statistic register.

26.2.4.7.1 TCP/UDP/IP Header for the First Frames

The hardware makes the following changes to the headers of the first packet that is derived from each TCP segmentation context.

MAC Header (for SNAP)

- Type/Len field = MSS + MACLEN + IPLEN + L4LEN - 14 -4 (if VLAN added by Software)

Ipv4 Header

- IP Identification: Value in the IPv4 header of the prototype header in the packet data itself
- IP Total Length = MSS + L4LEN + IPLEN
- IP Checksum

Ipv6 Header

- Payload Length = MSS + L4LEN + IPV6_HDR_extension[1]

TCP Header

- Sequence Number: The value is the Sequence Number of the first TCP byte in this frame.
- The flag values of the first frame are set by ANDing the flag word in the pseudo header with the DTXTCPFLGL.TCP_flg_first_seg register field. The default value of the DTXTCPFLGL.TCP_flg_first_seg are set so that the FIN flag and the PSH flag are cleared in the first frame.
- TCP Checksum

UDP Header

- UDP Length = MSS + L4LEN
- UDP Checksum

26.2.4.7.2 TCP/IP Header for the Subsequent Frames

The hardware makes the following changes to the headers for subsequent packets that are derived as part of a TCP segmentation context:

Number of bytes left for transmission = PAYLEN - (N * MSS). Where N is the number of frames that have been transmitted.

MAC Header (for SNAP Packets)

Type/Len field = MSS + MACLEN + IPLEN + L4LEN - 14 - 4 (if VLAN added by Software)

Ipv4 Header

- IP Identification: incremented from last value (wrap around based on 16 bit-width))
- IP Total Length = MSS + L4LEN + IPLEN
- IP Checksum

Ipv6 Header

- Payload Length = MSS + L4LEN + IPV6_HDR_extension[1]

TCP Header

- Sequence Number update: Add previous TCP payload size to the previous sequence number value. This is equivalent to adding the MSS to the previous sequence number.
- The flag values of the subsequent frames are set by ANDing the flag word in the pseudo header with the DTXTCPFLGL.TCP_Flg_mid_seg register field. The default value of the DTXTCPFLGL.TCP_Flg_mid_seg are set so that if the FIN flag and the PSH flag are cleared in these frames.
- TCP Checksum

TCP Header

- UDP Length = MSS + L4LEN
- UDP Checksum

26.2.4.7.3 TCP/IP Header for the Last Frame

The hardware makes the following changes to the headers for the last frame of a TCP segmentation context:

1. IPV6_HDR_extension is calculated as IPLEN - 40 bytes.

1. IPV6_HDR_extension is calculated as IPLEN - 40 bytes.

Last frame payload bytes = PAYLEN - (N * MSS)

MAC Header (for SNAP Packets)

- Type/Len field = Last frame payload bytes + MACLEN + IPLEN + L4LEN - 14 - 4 (if VLAN added by Software)

Ipv4 Header

- IP Total Length = last frame payload bytes + L4LEN + IPLEN
- IP Identification: incremented from last value (wrap around based on 16 bit-width)
- IP Checksum

Ipv6 Header

- Payload Length = last frame payload bytes + L4LEN + IPV6_HDR_extension[1]

TCP Header

- Sequence Number update: Add previous TCP payload size to the previous sequence number value. This is equivalent to adding the MSS to the previous sequence number.
- The flag values of the last frames are set by ANDing the flag word in the pseudo header with the DTXTCPFLGH.TCP_Flg_lst_seg register field. The default value of the DTXTCPFLGH.TCP_Flg_lst_seg are set so that if the FIN flag and the PSH flag are set in the last frame.
- TCP Checksum

TCP Header

- UDP Length = Last frame payload bytes + L4LEN
- UDP Checksum

26.2.4.8 IP/TCP/UDP Checksum Offloading

The Controller performs checksum off loading as part of the TCP segmentation off-load feature.

These specific checksum are supported under TCP segmentation:

- IPv4 checksum
- TCP checksum

See Section 26.2.5 for description of checksum off loading of a single-send packet.

26.2.4.9 Data Flow

The flow used by the Controller to do TCP segmentation is as follows:

1. Get a descriptor with a request for a TSO off-load of a TCP packet.
2. First Segment processing:
   a. Fetch all the buffers containing the header as calculated by the *MACLEN*, *IPLEN* and *L4LEN* fields. Save the addresses and lengths of the buffers containing the header (up to 4 buffers). The header content is not saved.
   b. Fetch data up to the MSS from subsequent buffers & calculate the adequate checksum(s).
   c. Update the Header accordingly and update internal state of the packet (next data to fetch and TCP SN).

d. Send the packet to the network.

e. If total packet was sent, go to step 4. else continue.

3. Next segments

a. Wait for next arbitration of this queue.

b. Fetch all the buffers containing the header from the saved addresses. Subsequent reads of the header might be done with a no snoop attribute.

c. Fetch data up to the MSS or end of packet from subsequent buffers & calculate the adequate checksum(s).

d. Update the Header accordingly and update internal state of the packet (next data to fetch and TCP SN).

e. If total packet was sent, request is done, else restart from step 3.

4. Release all buffers (update head pointer).

Note: Descriptors are fetched in a parallel process according to the consumption of the buffers.

26.2.5 Checksum Offloading in Non-Segmentation Mode

The previous section on TCP Segmentation off-load describes the IP/TCP/UDP checksum off loading mechanism used in conjunction with TCP Segmentation. The same underlying mechanism can also be applied as a standalone feature. The main difference in normal packet mode (non-TCP Segmentation) is that only the checksum fields in the IP/TCP/UDP headers need to be updated.

Before taking advantage of the Controller's enhanced checksum off-load capability, a checksum context must be initialized. For the normal transmit checksum off-load feature this is performed by providing the device with a Descriptor with *TSE*=0b in the *TDESD.DCMD* field and setting either the *TXSM* or *IXSM* bits in the *TDESD.POPTS* field. Setting TSE=0b indicates that the normal checksum context is being set, as opposed to the segmentation context. For additional details on contexts, see Section 26.2.2.4.

Note: Enabling the checksum off loading capability without first initializing the appropriate checksum context leads to unpredictable results. CRC appending (TDESC.CMD.IFCS) must be enabled in TCP/IP checksum mode, since CRC must be inserted by hardware after the checksum has been calculated.

As mentioned in Section 26.2.2, it is not necessary to set a new context for each new packet. In many cases, the same checksum context can be used for a majority of the packet stream. In this case, some performance can be gained by only changing the context on an as needed basis or electing to use the off-load feature only for a particular traffic type, thereby avoiding the need to read all context descriptors except for the initial one.

Each checksum operates independently. Insertion of the IP and TCP checksum for each packet are enabled through the Transmit Data Descriptor POPTS.TSXM and POPTS.IXSM fields, respectively.

26.2.5.1 IP Checksum

Three fields in the Transmit Context Descriptor set the context of the IP checksum off loading feature:

- TUCMD.IPv4
- IPLEN
- MACLEN

TUCMD.IPv4=1b specifies that the packet type for this context is IPv4, and that the IP header checksum should be inserted. TUCMD.IPv4=0b indicates that the packet type is IPv6 (or some other protocol) and that the IP header checksum should not be inserted.

MACLEN specifies the byte offset from the start of the DMA'd data to the first byte to be included in the checksum, the start of the IP header. The minimal allowed value for this field is 12. The maximum value for this field is 127. This is adequate for typical applications.

Note: The MACLEN+IPLEN value needs to be less than the total DMA length for a packet. If this is not the case, the results are unpredictable.

IPLEN specifies the IP header length. Maximum allowed value for this field is 511 Bytes.

MACLEN+IPLEN specify where the IP checksum should stop. This is limited to the first 127+511 bytes of the packet and must be less than or equal to the total length of a given packet. If this is not the case, the result is unpredictable.

The 16-bit IPv4 Header Checksum is placed at the two bytes starting at MACLEN+10.

As mentioned in Section 26.2.2.2, Transmit Contexts, it is not necessary to set a new context for each new packet. In many cases, the same checksum context can be used for a majority of the packet stream. In this case, some performance can be gained by only changing the context on an as needed basis or electing to use the off-load feature only for a particular traffic type, thereby avoiding all context descriptor reads except for the initial one.

26.2.5.2 TCP Checksum

Three fields in the Transmit Context Descriptor set the context of the TCP checksum off loading feature:

- MACLEN
- IPLEN
- TUCMD.L4T

TUCMD.L4T=01b specifies that the packet type is TCP, and that the 16-bit TCP header checksum should be inserted at byte offset MACLEN+IPLEN+16. TUCMD.L4T=00b indicates that the packet is UDP and that the 16-bit checksum should be inserted starting at byte offset MACLEN+IPLEN+6.

IPLEN+MACLEN specifies the byte offset from the start of the DMA'd data to the first byte to be included in the checksum, the start of the TCP header. The minimal allowed value for this sum is 32/42 for UDP or TCP respectively.

Note: The IPLEN+MACLEN+L4LEN value needs to be less than the total DMA length for a packet. If this is not the case, the results are unpredictable.

The TCP/UDP checksum always continues to the last byte of the DMA data.

Note: For non-TSO, software still needs to calculate a full checksum for the TCP/UDP pseudo-header. This checksum of the pseudo-header should be placed in the packet data buffer at the appropriate offset for the checksum calculation.

26.2.5.3 SCTP CRC Offloading

For SCTP packets, a CRC32 checksum offload is provided.

Three fields in the Transmit Context Descriptor set the context of the STCP checksum off loading feature:

- MACLEN
- IPLEN
- TUCMD.L4T

TUCMD.L4T=10b specifies that the packet type is SCTP, and that the 32-bit STCP CRC should be inserted at byte offset MACLEN+IPLEN+8.

IPLEN+MACLEN specifies the byte offset from the start of the DMA'd data to the first byte to be included in the checksum, the start of the STCP header. The minimal allowed value for this sum is 26.

The SCTP CRC calculation always continues to the last byte of the DMA data.

The SCTP total L3 payload size (PAYLEN - IPLEN - MACLEN) should be a multiple of 4 bytes (SCTP padding not supported).

1. TSO is not available for SCTP packets.
2. The CRC field of the SCTP header must be set to zero prior to requesting a CRC calculation offload.

26.2.5.4 Checksum Supported Per Packet Types

The following table summarizes which checksum is supported per packet type.

Note: TSO is not supported for packet types for which IP checksum & TCP checksum can not be calculated.

Table 26-45. Checksum Per Packet Type

Packet Type	Hardware IP Checksum Calculation	Hardware TCP/UDP/SCTP Checksum Calculation
Ipv4 packets	Yes	Yes
Ipv6 packets	No (n/a)	Yes
Ipv6 packet with next header options: • Hop-by-Hop options • Destinations options • Routing (w len 0b) • Routing (w len >0b) • Fragment • Home option	 No (n/a) No (n/a) No (n/a) No (n/a) No (n/a) No (n/a)	 Yes Yes Yes No No No
Ipv4 tunnels: • Ipv4 packet in an Ipv4 tunnel • Ipv6 packet in an Ipv4 tunnel	 Either IP or TCP/SCTP[1] Either IP or TCP/SCTP[1]	 Either IP or TCP/SCTP [1] Either IP or TCP/SCTP[1]
Ipv6 tunnels: • Ipv4 packet in an Ipv6 tunnel • Ipv6 packet in an Ipv6 tunnel	 No No	 Yes Yes
Packet is an Ipv4 fragment	Yes	No
Packet is greater than 1518, 1522 or 1526 bytes; (LPE=1b)	Yes	Yes
Packet has 802.3ac tag	Yes	Yes
Packet has TCP or UDP options	Yes	Yes
IP header's protocol field contains protocol # other than TCP or UDP.	Yes	No

1. For the tunneled case, the driver might do only the TCP checksum or Ipv4 checksum. If TCP checksum is desired, the driver should define the IP header length as the combined length of both IP headers in the packet. If an IPv4 checksum is required, the IP header length should be set to the Ipv4 header length.

26.2.6 Multiple Transmit Queues

The number of transmit queues is 16, to match the expected number of CPU cores on server processors and to support virtualization mode.

If there are more CPUs cores than queues, then one queue might be used to service more than one CPU. For transmission process, each thread might place a queue in the host memory of the CPU it is tied to.

26.3 Interrupts

26.3.1 Mapping of Interrupt Causes

The Controller supports the following interrupt modes:

- PCI legacy interrupts or MSI - selected when GPIE.Multiple_MSIX is 0b
- MSI-X when GPIE.Multiple_MSIX is 1b.

Note: If only one MSI-X vector is allocated by the operating system, then the driver might use the non MSI-X mapping method even in MSI-X mode.

Mapping of interrupts causes is different in each of the above modes and is described below.

26.3.1.1 Legacy and MSI Interrupt Modes

In legacy and MSI modes, an interrupt cause is reflected by setting a bit in the EICR register. This section describes the mapping of interrupt causes (a specific Rx queue event or a Link Status Change event) to bits in the EICR.

Mapping of queue-related causes is accomplished through the IVAR register. Each possible queue interrupt cause (each Rx or Tx queue) is allocated an entry in the IVAR, and each entry in the IVAR identifies one bit in the EICR register among the bits allocated to queue interrupt causes. It is possible to map multiple interrupt causes into the same *EICR* bit.

In this mode, different queue related interrupt causes can be mapped to the first 16 bits of the EICR register.

The following configuration and parameters are involved:

- The IVAR[3:0] entries map 8 Tx queues and 16 Rx queues into the EICR[7:0] bits.
- The IVAR_MISC that maps non-queue causes is not used.
- The *EICR[30]* bit is allocated to the TCP timer interrupt cause.
- The *EICR[31]* bit is allocated to the other interrupt causes summarized in the ICR register.
- A single interrupt vector is provided.

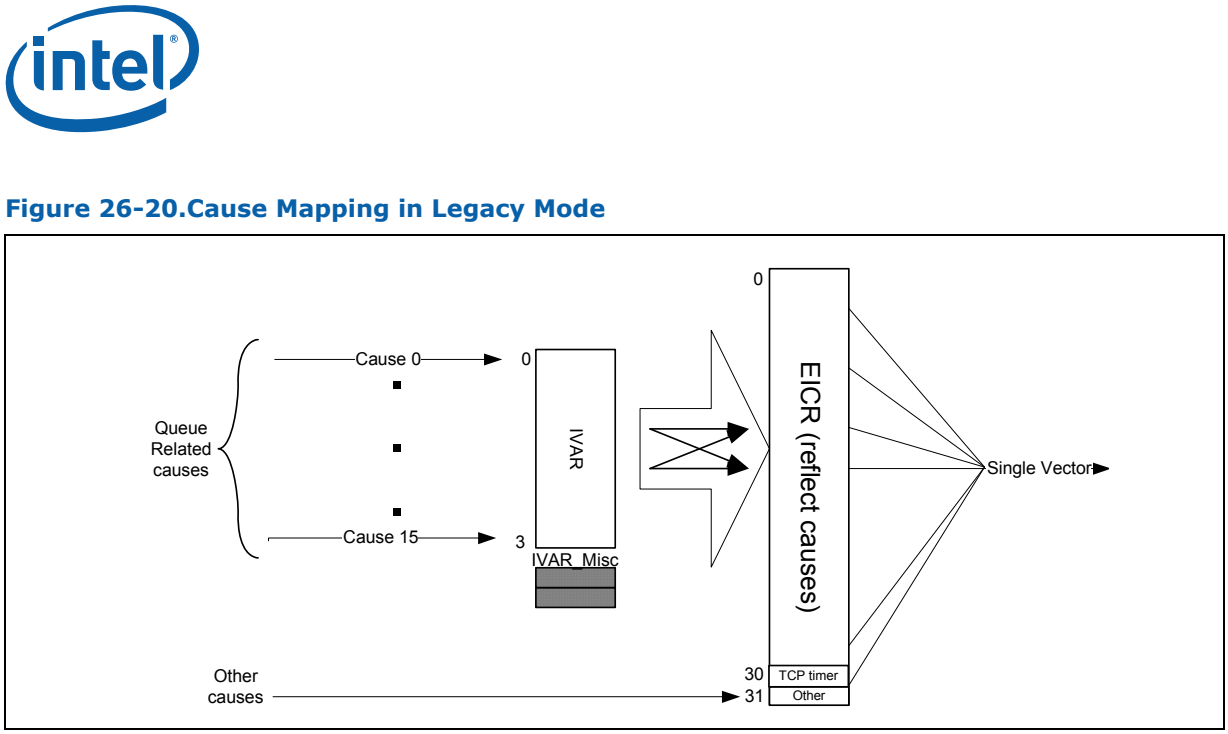


Figure 26-20.Cause Mapping in Legacy Mode

The table below maps the different interrupt causes into the IVAR registers.

Table 26-46. Cause Allocation in the IVAR Registers - MSI and Legacy Mode

Interrupt	Entry	Description
Rx_i	i* (i= 0...7)	Receive queues i - Associates an interrupt occurring in the Rx queues i with a corresponding bit in the EICR register.
Tx_i	i*+1 (i= 0...7)	Transmit queues i- Associates an interrupt occurring in the Tx queues I with a corresponding bit in the EICR register.

26.3.1.2 MSI-X Mode - VMDq Mode

In a VMDq setup, the Controller can request up to 25 Vectors.

In MSI-X mode, an interrupt cause is mapped into an MSI-X vector. This section describes the mapping of interrupt causes (a specific Rx queue event or other events) to MSI-X vectors.

Mapping is accomplished through the IVAR register. Each possible cause for an interrupt is allocated an entry in the IVAR, and each entry in the IVAR identifies one MSI-X vector. It is possible to map multiple interrupt causes into the MSI-X vector.

The EICR also reflects interrupt vectors. The EICR bits allocated for queue causes reflect the MSI-X vector (bit 2 is set when MSI-X vector 2 is used). The following configuration and parameters are involved:

- The IVAR[3:0] registers map 16 Tx queues, 16 Rx queues, events to up to 8 interrupt vectors.
- The IVAR_MISC register maps a TCP timer and other events to 2 MSI-X vectors.

Figure 26-21. Cause Mapping in MSI-X Mode

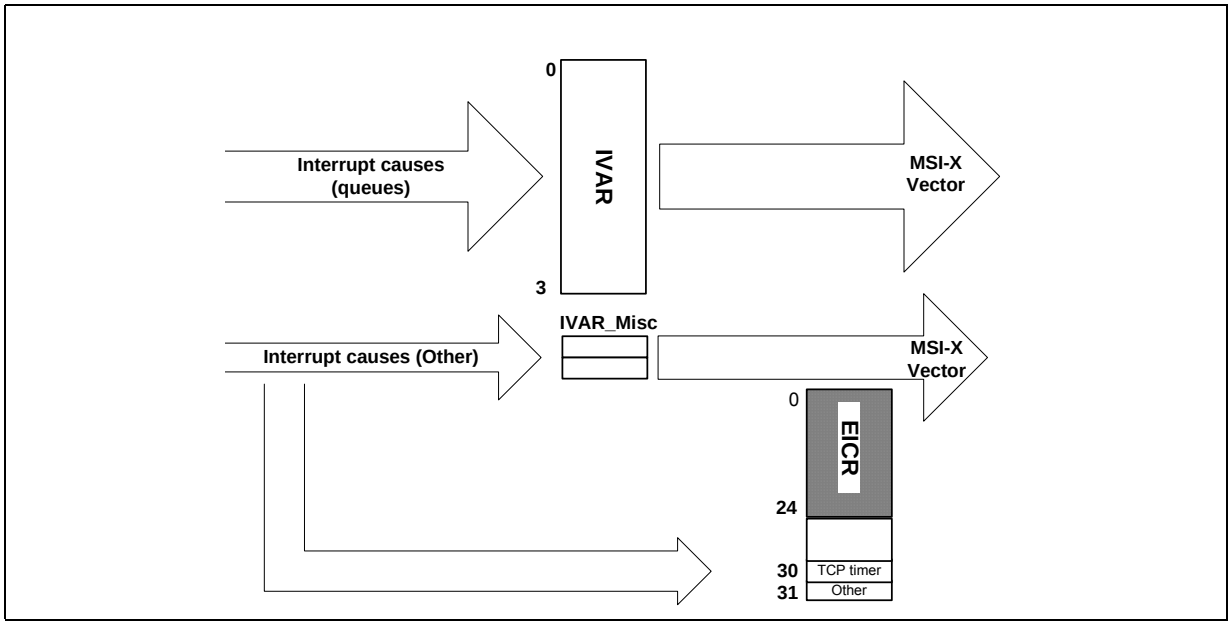


Table 26-47 below defines which interrupt cause is represented by each entry in the MSI-X Allocation registers. The software has access to 18 mapping entries to map each cause to one of the 25 MSI-x vectors.

Table 26-47. Cause Allocation in the IVAR Registers

Interrupt	Entry	Description
Rx_i	i* (i= 0...7)	Receive queues i - Associates an interrupt occurring in the Rx queues i with a corresponding entry in the MSI-X Allocation registers.
Tx_i	i*+1 (i= 0...7)	Transmit queues i- Associates an interrupt occurring in the Tx queues I with a corresponding entry in the MSI-X Allocation registers.
TCP timer		TCP Timer - Associates an interrupt issued by the TCP timer with a corresponding entry in the MSI-X Allocation registers
Other cause		Other causes - Associates an interrupt issued by the "other causes" with a corresponding entry in the MSI-X Allocation registers

26.3.2 Legacy Interrupt Registers

The interrupt logic consists of the registers listed in the tables below, plus the registers associated with MSI/MSI-X signaling. The first table describes the use of the registers in legacy mode and the second one the use of the register when using the extended interrupts functionality.

Table 26-48. Interrupt Registers - Legacy Mode

Register	Acronym	Function
Interrupt Cause	ICR	Records interrupt conditions.
Interrupt Cause Set	ICS	Allows software to set bits in the ICR.
Interrupt Mask Set/Read	IMS	Sets or reads bits in the interrupt mask.
Interrupt Mask Clear	IMC	Clears bits in the interrupt mask.
Interrupt Acknowledge auto-mask	IAM	Under some conditions, the content of this register is copied to the mask register following read or write of ICR.

Table 26-49. Interrupt Registers - Extended Mode

Register	Acronym	Function
Extended Interrupt Cause	EICR	Records interrupt causes from receive and transmit queues. An interrupt is signaled when unmasked bits in this register are set.
Extended Interrupt Cause Set	EICS	Allows software to set bits in the Interrupt Cause register.
Extended Interrupt Mask Set/Read	EIMS	Sets or read bits in the interrupt mask.
Extended Interrupt Mask Clear	EIMC	Clears bits in the interrupt mask.
Extended Interrupt Auto Clear	EIAC	Allows bits in the EICR to be cleared automatically following an MSI-X interrupt without a read or write of the EICR.
Extended Interrupt Acknowledge auto-mask	EIAM	This register is used to decide which masks are cleared in the extended mask register following read or write of EICR or which masks are set following a write to EICS. In MSI-X mode, this register also controls which bits in EIMC are cleared automatically following an MSI-X interrupt.
Interrupt Cause	ICR	Records interrupt conditions for special conditions - a single interrupt from all the conditions of ICR is reflected in the "other" field of the EICR.
Interrupt Cause Set	ICS	Allows software to set bits in the ICR.
Interrupt Mask Set/Read	IMS	Sets or reads bits in the other interrupt mask.

Table 26-49. Interrupt Registers - Extended Mode (Continued)

Interrupt Mask Clear	IMC	Clears bits in the Other interrupt mask.
Interrupt Acknowledge auto-mask	IAM	Under some conditions, the content of this register is copied to the mask register following read or write of ICR.
General Purpose Interrupt Enable	GPIE	Controls different behaviors of the interrupt mechanism.

26.3.2.1 Interrupt Cause Register (ICR)

26.3.2.1.1 Legacy Mode

In Legacy mode, ICR is used as the sole interrupt cause register. Upon reception of an interrupt, the interrupt handling routine can read this register in order to find out what are the causes of this interrupt.

26.3.2.1.2 Advanced Mode

In advanced mode, this register captures the interrupt causes not directly captured by the EICR. These are infrequent management interrupts and error conditions.

When EICR is used in advanced mode, the RX /TX related bits in ICR should be masked.

ICR bits are cleared on register read. If GPIE.NSICR = 0b, then the clear on read occurs only if no bit is set in the IMS or at least one bit is set in the IMS and there is a true interrupt as reflected in ICR.INTA.

26.3.2.2 Interrupt Cause Set Register (ICS)

This registers allows setting the bits of ICR by software, by writing a 1b in the corresponding bits in ICS. Used usually to rearm interrupts the software didn't have time to handle in the current interrupt routine.

26.3.2.3 Interrupt Mask Set/Read Register (IMS)

An interrupt is enabled if its corresponding mask bit in this register is set to 1b, and disabled if its corresponding mask bit is set to 0b. A PCIe* interrupt is generated whenever one of the bits in this register is set, and the corresponding interrupt condition occurs. The occurrence of an interrupt condition is reflected by having a bit set in the Interrupt Cause Register.

Reading this register returns which bits have an interrupt mask set.

A particular interrupt might be enabled by writing a 1b to the corresponding mask bit in this register. Any bits written with a 0b are unchanged. Thus, if software desires to disable a particular interrupt condition that had been previously enabled, it must write to the Interrupt Mask Clear Register (see below), rather than writing a 0b to a bit in this register.

26.3.2.4 Interrupt Mask Clear Register (IMC)

Software blocks interrupts by clearing the corresponding mask bit. This is accomplished by writing a 1b to the corresponding bit in this register. Bits written with 0b are unchanged (their mask status does not change).

26.3.2.5 Interrupt Acknowledge Auto-Mask Register (IAM)

An ICR read or write has the side effect of writing the contents of this register to the mask register. If GPIE.NSICR = 0b, then the copy of this register to the mask register occurs only if at least one bit is set in the mask register and there is a true interrupt as reflected in ICR.INTA.

26.3.2.6 Extended Interrupt Cause Registers (EICR)

26.3.2.6.1 MSI/INT-A Mode (GPIE.Multiple_MSIX = 0)

This register records the interrupts causes to provide to the software information on the interrupt source.

The interrupt causes include:

1. The Receive and Transmit queues — Each queue (either Tx or Rx) can be mapped to one of the 16 interrupt causes bits (RxTxQ) available in this register according to the mapping in the IVAR registers
2. Indication for the TCP timer interrupt.

— Writing a 1b clears the corresponding bit in this register. Reading this register auto-clears all bits.

26.3.2.6.2 MSI-X Mode (GPIE.Multiple_MSIX = 1)

This register records the interrupt vectors currently emitted. In this mode only the first 25 bits are valid.

For all the subsequent registers, in MSI-X mode, each bit controls the behavior of one vector.

Bits in this register can be configured to auto-clear when the MSI-X interrupt message is sent, in order to minimize driver overhead when using MSI-X interrupt signaling.

Writing a 1b clears the corresponding bit in this register. Reading this register does not clear any bits.

26.3.2.7 Extended Interrupt Cause Set Register (EICS)

This registers allows to set the bits of EICR by software, by writing a 1b in the corresponding bits in EICS. Used usually to rearm interrupts that the software didn't have time to handle in the current interrupt routine.

26.3.2.8 Extended Interrupt Mask Set and Read Register (EIMS) & Extended Interrupt Mask Clear Register (EIMC)

Interrupts appear on PCIe* only if the interrupt cause bit is a one and the corresponding interrupt mask bit is a one. Software blocks assertion of an interrupt by clearing the corresponding bit in the mask register. The cause bit stores the interrupt event regardless of the state of the mask bit. Different Clear (EIMC) and set (EIMS) registers make this register more “thread safe” by avoiding a read-modify-write operation on the mask register. The mask bit is set for each bit written to a one in the set register (EIMS) and cleared for each bit written in the clear register (EIMC). Reading the set register (EIMS) returns the current mask register value.

26.3.2.9 Extended Interrupt Auto Clear Enable Register (EIAC)

Each bit in this register enables clearing of the corresponding bit in EICR following interrupt generation. When a bit is set, the corresponding bit in EICR are automatically cleared following an interrupt. This feature should only be used in MSI-X mode.

When used in conjunction with MSI-X interrupt vector, this feature allows interrupt cause recognition, and selective interrupt cause, without requiring software to read or write the EICR register; therefore, the penalty related to a PCIe* read or write transaction is avoided.

See section 26.3.5 for additional information on the interrupt cause reset process.

26.3.2.10 Extended Interrupt Auto Mask Enable Register (EIAM)

Each bit set in this register enables clearing of the corresponding bit in the extended mask register following read or write-to-clear to EICR. It also enables setting of the corresponding bit in the extended mask register following a write-to-set to EICS.

This mode is provided in case MSI-X is not used, and therefore auto-clear through EIAC register is not available.

In MSI-X mode, the driver software might set the bits of this register to select mask bits that must be reset during interrupt processing. In this mode, each bit in this register enables clearing of the corresponding bit in EIMC following interrupt generation.

26.3.2.11 GPIE Register

There are a few bits in the GPIE register that define the behavior of the interrupt mechanism. The setting of these bits is different in each mode of operation. The following table describes the recommended setting of these bits in the different modes:

Table 26-50. Settings for Different Interrupt Modes

Field	Bit(s)	Initial Value	Description	INT-x/ MSI + Legacy	INT-x/ MSI + Extend	MSI-X Multi vector	MSI-X Single vector
Multiple_ MSIX	4	0b	**Multiple_MSIX - multiple vectors:** 0b = non-MSIX or MSI-X with 1 vector IVAR maps Rx/Tx causes to 16 EICR bits, but MSIX[0] is asserted for all. 1b = MSIX mode, IVAR maps Rx/Tx causes to 25 EICR bits. When set, the EICR register is not clear on read.	0b	0b	1b	0b
EIAME	30	0b	**EIAME:** When set, upon firing of an MSI-X message, mask bits set in EIAM associated with this message are cleared. Otherwise, EIAM is used only upon read or write of EICR/EICS registers.	0b	0b	1b	1b
PBA_ support	31	0b	**PBA support:** When set, setting one of the extended interrupts masks via EIMS causes the PBA bit of the associated MSI-X vector to be cleared. Otherwise, the Controller behaves in a way that supports legacy INT-x interrupts. Should be cleared when working in INT-x or MSI mode and set in MSI-X mode.	0b	0b	1b	1b

26.3.3 MSI-X and Vectors

MSI-X defines a separate optional extension to basic MSI functionality. Compared to MSI, MSI-X supports a larger maximum number of vectors per function, the ability for software to control aliasing when fewer vectors are allocated than requested, plus the ability for each vector to use an independent address and data value, specified by a table that resides in Memory Space. However, most of the other characteristics of MSI-X are identical to those of MSI. For more information on MSI-X, see the PCI Local Bus Specification, Revision 3.0.

MSI-X maps each of the the Controller interrupt causes into an interrupt vector that is conveyed by the Controller as a posted-write PCIe* transaction. Mapping of an interrupt cause into an MSI-X vector is determined by system software (a device driver) through a translation table stored in the MSI-X Allocation registers. Each entry of the allocation registers defines the vector for a single interrupt cause.

There are 18 extended interrupt causes that exist in the Controller:

1. traffic causes — 16 Tx, 16 Rx.
2. TCP timer

26.3.4 Interrupt Moderation

An interrupt is generated upon receiving of incoming packets, as throttled by the EITR registers (see Section 28.7.1.22). There is an EITR register per MSI-X vector.

In MSI-X mode, each active bit in EICR can trigger the interrupt vector it is allocated to. Following the allocation, the EITR corresponding to the MSI-X vector is tied to one or more bits in EICR.

When MSI-X is not activated, the interrupt moderation is controlled by register EITR[0].

Software can use EITR to limit the rate of delivery of interrupts to the host CPU. This register provides a guaranteed inter-interrupt delay between interrupts asserted by the network controller, regardless of network traffic conditions.

The following formula converts the inter-interrupt interval value to the common 'interrupts/sec.' performance metric:

$$\text{interrupts/sec} = (2 * 10^{-6}\text{sec} \times \text{interval})^{-1}$$

Note: In the Controller the interval granularity is 2 μsec so some of the LSB bits of the interval are used for the low latency interrupt moderation.

For example, if the interval is programmed to 125d, the network controller guarantees the CPU is not interrupted by the network controller for at least 250 μs from the last interrupt. In this case, the maximum observable interrupt rate from the adapter should not exceed 4000 interrupts/sec.

Inversely, inter-interrupt interval value can be calculated as:

$$\text{inter-interrupt interval} = (2 * 10^{-6}\ \text{sec} \times \text{interrupt/sec})^{-1}$$

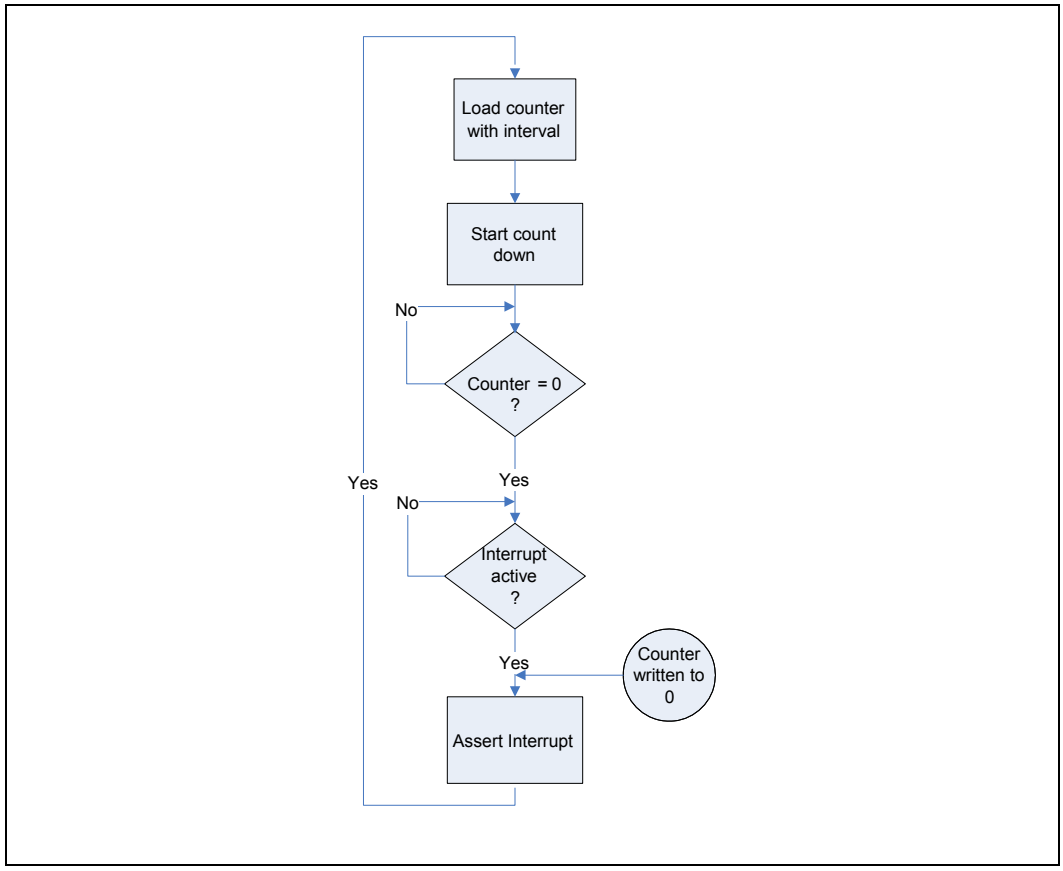
The optimal performance setting for this register is system and configuration specific.

The Extended Interrupt Throttle Register should default to zero upon initialization and reset. It loads in the value programmed by the software after software initializes the device.

When software wants to force an immediate interrupt, for example after setting a bit in the EICR with the Extended Interrupt Cause Set register, a value of 0 can be written to the Counter to generate an interrupt immediately. This write should include re-writing the *Interval* field with the desired constant, as it is used to reload the Counter immediately for the next throttling interval.

The Controller implements interrupt moderation to reduce the number of interrupts software processes. The moderation scheme is based on the EITR (Interrupt Throttle Register). Whenever an interrupt event happens, the corresponding bit in the EICR is activated. However, an interrupt message is not sent out on the PCIe* interface until the EITR counter assigned to that *EICR* bit has counted down to zero. As soon as the interrupt is issued, the EITR counter is reloaded with its initial value and the process repeats again. The interrupt flow should follow the diagram below:

Figure 26-22.Interrupt Throttle Flow Diagram



For cases where the Controller is connected to a small number of clients, it is desirable to fire off the interrupt as soon as possible with minimum latency. For these cases, when the EITR counter counts down to zero and no interrupt event has happened, then the EITR counter is not re-armed but stays at zero. Thus, the next interrupt event triggers an interrupt immediately. That scenario is illustrated as "Case B" below.

Figure 26-23.Case A: Heavy Load, Interrupts Moderated

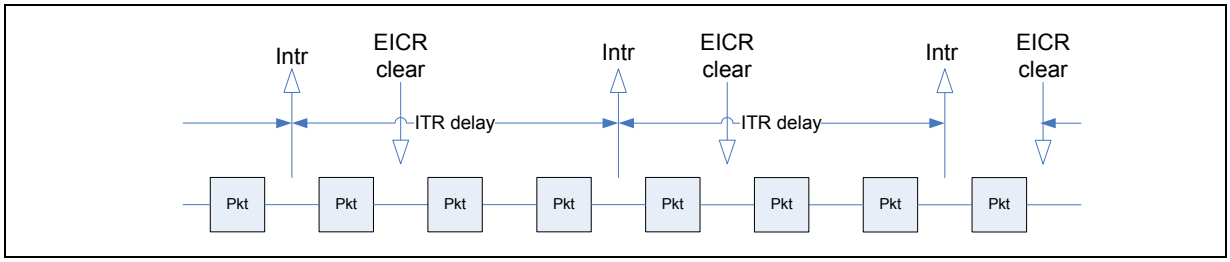
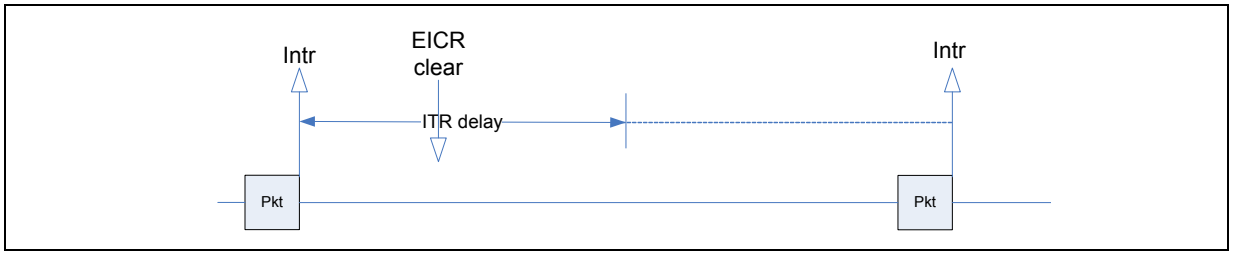


Figure 26-24.Light Load, Interrupts Immediately on Packet Receive



26.3.5 Clearing Interrupt Causes

The Controller has three methods available for to clear EICR bits: Autoclear, clear-on-write, and clear-on-read. ICR bits might only be cleared with clear-on-write or clear-on-read.

26.3.5.1 Auto-Clear

In systems that support MSI-X, the interrupt vector allows the interrupt service routine to know the interrupt cause without reading the EICR. With interrupt moderation active, software load from spurious interrupts is minimized. In this case, the software overhead of a I/O read or write can be avoided by setting appropriate EICR bits to autoclear mode by setting the corresponding bits in the Extended Interrupt Auto-clear Enable Register (EIAC).

When auto-clear is enabled for an interrupt cause, the *EICR* bit is set when a cause event mapped to this vector occurs. When the EITR Counter reaches zero, the MSI-X message is sent on PCIe*. Then the *EICR* bit is cleared and enabled to be set by a new cause event. The vector in the MSI-X message signals software the cause of the interrupt to be serviced.

It is possible that in the time after the *EICR* bit is cleared and the interrupt service routine services the cause, for example checking the transmit and receive queues, that another cause event occurs that is then serviced by this ISR call, yet the *EICR* bit remains set. This results in a "spurious interrupt". Software can detect this case, for example if there are no entries that require service in the transmit and receive queues, and exit knowing that the interrupt has been automatically cleared. The use of interrupt moderations through the EITR register limits the extra software overhead that can be caused by these spurious interrupts.

26.3.5.2 Write to Clear

In the case where the driver wishes to configure itself in MSI-X mode to not use the "auto-clear" feature, it might clear the EICR bits by writing to the EICR register. Any bits written with a 1b is cleared. Any bits written with a 0b remain unchanged.

26.3.5.3 Read to Clear

The EICR and ICR registers are cleared on a read.

The driver should never do a read-to-clear of the EICR when in MSI-X mode, since this might clear interrupt cause events which are processed by a different interrupt handler (assuming multiple vectors).

26.3.6 Rate Controlled Low Latency Interrupts (LLI)

There are some types of network traffic for which latency is a critical issue. For these types of traffic, interrupt moderation hurts performance by increasing latency between the time a packet is received by hardware and the time it is handled to the host operating system. This traffic can be identified by the value, in conjunction with Control Bits and specific size. In addition packets with specific ethernet types, TCP flag or specific VLAN priority might generate an immediate interrupt.

Low latency interrupts shares the filters used by the queueing mechanism described in Section 26.1.1. Each of these filters, in addition to the queueing action might also indicate matching packets might generate immediate interrupt.

If a received packet matches one of these filters, hardware should interrupt immediately, overriding the interrupt moderation by the EITR counter.

Each time a Low Latency Interrupt is fired, the EITR interval is loaded and down-counting starts again.

The logic of the low latency interrupt mechanism is as follows:

- There are 8 filters. The content of each filter is described in Section 26.1.1.5. The immediate interrupt action of each filter can be enabled or disabled. If one of the filters detects an adequate packet, an immediate interrupt is issued.
- There are 8 flex filters. The content of each filter is described in Section 26.1.1.6. The immediate interrupt action of each filter can be enabled or disabled. If one of the filters detects an adequate packet, an immediate interrupt is issued.
- When VLAN priority filtering is enabled, VLAN packets must trigger an immediate interrupt when the VLAN Priority is equal to or above the VLAN priority threshold. This is regardless of the status of the or Flex filters.
- The SYN packets filter defined in Section 26.1.1.7 and the ethernet type filters defined in section Section 26.1.1.4 might also be used to indicate low latency interrupt conditions.

Note: EITR is re-armed to the 'EITR.Moderation Counter' value following a low latency interrupt.

- Immediate interrupts are available only when using advanced receive descriptors and not for legacy descriptors.
- Packets that are dropped or have errors do not cause a Low Latency Interrupt.

26.3.6.1 Rate Control Mechanism

In a network with lots of latency sensitive traffics the Low Latency Interrupt can eliminate the Interrupt throttling capability by flooding the Host with too many interrupts (more than the Host can handle).

In order to mitigate the above, the Controller supports a credit base mechanism to control the rate of the Low Latency Interrupts.

Rules:

- The default value of each counter is 0b (no moderation). This also preserves backward compatibility.
- The counter increments at a configurable rate, and saturates at the maximum value (31d).
  - The configurable rate granularity is 4 μs (250K interrupt/sec. down to 250K/32 ~ 8K interrupts per sec.).
- A LLI might be issued as long as the counter value is strictly positive (> zero).
  - The credit counter allows bursts of low latency interrupts but the interrupt average are not more than the configured rate.
- Each time a Low Latency Interrupt is fired the credit counter decrements by one.
- Once the counter reaches zero, a low latency interrupt cannot be fired
  - Must wait for the next ITR expired or for the next incrementing of this counter (if the EITR expired happened first the counter does not decrement).

The EITR and GPIE registers manage rate control of LLI:

- The *LL Interval* field in the GPIE register controls the rate of credits
- The 5-bit *LL Counter* field in the EITR register contains the credits

26.3.7 TCP Timer Interrupt

26.3.7.1 Introduction

In order to implement TCP timers for IOAT, software needs to take action periodically (every 10 milliseconds). Today, the driver must rely on software-based timers, whose granularity can change from platform to platform. This software timer generates a software NIC interrupt, which then allows the driver to perform timer functions as part of its usual DPC, avoiding cache thrash and enabling parallelization. The timer interval is system-specific.

It would be more accurate and more efficient for this periodic timer to be implemented in hardware. The driver would program a timeout value (usual value of 10 ms), and each time the timer expires, hardware sets a specific bit in the EICR. When an interrupt occurs (due to normal interrupt moderation schemes), software reads the EICR and discovers that it needs to process timer events during that DPC.

The timeout should be programmable by the driver, and the driver should be able to disable the timer interrupt if it is not needed.

26.3.7.2 Description

A stand-alone down-counter is implemented. An interrupt is issued each time the value of the counter is zero.

The software is responsible for setting initial value for the timer in the *TCPTIMER.Duration* field. Kick-starting is done by writing a 1b to the *TCPTIMER.KickStart* bit.

Following the kick-start, an internal counter is set to the value defined by the *TCPTIMER.Duration* field. Then during the count operation, the counter is decreased by one each millisecond. When the counter reaches zero, an interrupt is issued (see EICR register Section 28.7.1.1). The counter re-starts counting from its initial value if the *TCPTIMER.Loop* field is set.

26.3.8 Special Cases About Interrupts

26.3.8.1 Ordering Issue Between CSR Rd/Wr Received With Error by the Controller in the PCH and its Generated Interrupt

When a CSR Rd/Wr request is received by the Controller in the PCH with error, interrupt is generated to EP before the request has been serviced by GbE for example read data returned to host or write data written to its destination. This breaks the PCIe* ordering rules as interrupt messaging may actually pass actual CSR read/write completion.

26.4 802.1q VLAN Support

The Controller provides several specific mechanisms to support 802.1q VLANs:

- Optional adding (for transmits) and stripping (for receives) of IEEE 802.1q VLAN tags.
- Optional ability to filter packets belonging to certain 802.1q VLANs.

26.4.1 802.1q VLAN Packet Format

The following diagram compares an untagged 802.3 Ethernet packet with an 802.1q VLAN tagged packet:

Table 26-51. Comparing Packets

802.3 Packet	#Octets		802.1q VLAN Packet	#Octets
DA	6		DA	6
SA	6		SA	6
Type/Length	2		802.1q Tag	4
Data	46-1500		Type/Length	2
CRC	4		Data	46-1500
			CRC*	4

Note: The CRC for the 802.1q tagged frame is re-computed, so that it covers the entire tagged frame including the 802.1q tag header. Also, max frame size for an 802.1q VLAN packet is 1522 octets as opposed to 1518 octets for a normal 802.3z Ethernet packet.

26.4.2 802.1q Tagged Frames

For 802.1q, the *Tag Header* field consists of four octets comprised of the Tag Protocol Identifier (TPID) and Tag Control Information (TCI); each taking 2 octets. The first 16 bits of the tag header makes up the TPID. It contains the “protocol type” which identifies the packet as a valid 802.1q tagged packet.

The two octets making up the TCI contain three fields:

- User Priority (UP)
- Canonical Form Indicator (CFI). Should be 0b for transmits. For receives, the device has the capability to filter out packets that have this bit set. See the CFIEN and CFI bits in the RCTL described in Section 28.9.1.1.
- VLAN Identifier (VID)

The bit ordering is shown below:

Table 26-52. TCI Bit Ordering

Octet 1								Octet 2							
UP				VID											

26.4.3 Transmitting and Receiving 802.1q Packets

26.4.3.1 Adding 802.1q Tags on Transmits

Software might command the Controller to insert an 802.1q VLAN tag on a per packet or per flow basis. If CTRL.VME is set to 1b, and the *VLE* bit in the transmit descriptor is set to 1b, then the Controller inserts a VLAN tag into the packet that it transmits over the wire. The *Tag Protocol Identifier (TPID)* field of the 802.1q tag comes from the VET register. 8021.Q tag insertion is done in different ways for legacy and advanced Tx descriptors:

- Legacy Transmit Descriptors:, The Tag Control Information (TCI) of the 802.1q tag comes from the *VLAN* field (see Figure 26-8) of the descriptor. See Table 26-26 for more information regarding hardware insertion of tags for transmits.
- Advanced Transmit Descriptor: The Tag Control Information (TCI) of the 802.1q tag comes from the VLAN *Tag* field (see Table 26.2.2.2.1) of the advanced context descriptor. The *IDX* field of the advanced Tx descriptor should be set to the adequate context.

26.4.3.2 Stripping 802.1q Tags on Receives

Software might instruct the Controller to strip 802.1q VLAN tags from received packets. If the *CTRL.VME* bit is set to 1b, and the incoming packet is an 802.1q VLAN packet (its *Ethernet Type* field matched the VET), then the Controller strips the 4 byte VLAN tag from the packet, and stores the TCI in the VLAN *Tag* field (see Figure 26-4 and See "Receive UDP Fragmentation Checksum) of the receive descriptor.

The Controller also sets the *VP* bit in the receive descriptor to indicate that the packet had a VLAN tag that was stripped. If the *CTRL.VME* bit is not set, the 802.1Q packets can still be received if they pass the receive filter, but the VLAN tag is not stripped and the *VP* bit is not set. See Table 26-25 for more information regarding receive packet filtering.

26.4.4 802.1q VLAN Packet Filtering

VLAN filtering is enabled by setting the *RCTL.VFE* bit to 1b. If enabled, hardware compares the type field of the incoming packet to a 16-bit field in the VLAN Ether Type (VET) register. If the VLAN type field in the incoming packet matches the VET register, the packet is then compared against the VLAN Filter Table Array (VFTA[127:0]) for acceptance.

The Controller provides exact VLAN filtering for VLAN tags for host traffic and VLAN tags for manageability traffic.

The *Virtual LAN ID* field indexes a 4096 bit vector. If the indexed bit in the vector is one; there is a Virtual LAN match. Software might set the entire bit vector to ones if the node does not implement 802.1q filtering. The register description of the VLAN Filter Table Array is described in detail in Section 28.9.1.20.

In summary, the 4096-bit vector is comprised of 128, 32-bit registers. The *VLAN Identifier (VID)* field consists of 12 bits. The upper 7 bits of this field are decoded to determine the 32-bit register in the VLAN Filter Table Array to address and the lower 5 bits determine which of the 32 bits in the register to evaluate for matching.

The BMC configures the Controller with eight different manageability VIDs via the Management VLAN TAG Value [7:0] - MAVTV[7:0] registers and enables each filter in the MDEF register.

Two other bits in the Receive Control register (see Section 28.9.1.1), CFIEN and CFI, are also used in conjunction with 802.1q VLAN filtering operations. CFIEN enables the comparison of the value of the *CFI* bit in the 802.1q packet to the Receive Control register *CFI* bit as acceptance criteria for the packet.

Note: The *VFE* bit does not effect whether the VLAN tag is stripped. It only effects whether the VLAN packet passes the receive filter.

The following table lists reception actions per control bit settings.

Figure 26-25.Packet Reception Decision Table

Is packet 802.1q?	CTRL. VME	RCTL. VFE	Action
No	X[1]	X[1]	Normal packet reception
Yes	0b	0b	Receive a VLAN packet if it passes the standard MAC address filters (only). Leave the packet as received in the data buffer. *VP* bit in receive descriptor is cleared.
Yes	0b	1b	Receive a VLAN packet if it passes the standard filters and the VLAN filter table. Leave the packet as received in the data buffer (the VLAN tag would not be stripped). *VP* bit in receive descriptor is cleared.
Yes	1b	0b	Receive a VLAN packet if it passes the standard filters (only). Strip off the VLAN information (four bytes) from the incoming packet and store in the descriptor. Sets *VP* bit in receive descriptor.
Yes	1b	1b	Receive a VLAN packet if it passes the standard filters and the VLAN filter table. Strip off the VLAN information (four bytes) from the incoming packet and store in the descriptor. Sets *VP* bit in receive descriptor.

1. X - Don't care

Note: A packet is defined as a VLAN/802.1q packet if its type field matches the VET.

26.4.5 Double VLAN Support

The Controller supports a mode where all received and sent packet have at least one VLAN tag in addition to the regular tagging which might optionally be added. This mode is used for systems where the switches add an additional tag containing switching information.

This mode is activated by setting *CTRL_EXT.EXTENDED_VLAN* bit. The default value of this bit is set according to the Ext_VLAN (bit 1) in the *Initialization Control 3* EEPROM word for ports 0 to 3. See Section 24.3.15 for more information.

The type of the VLAN tag used for the additional VLAN is defined in the *VET.VET_EXT* field.

26.4.5.1 Transmit Behavior With External VLAN

It is expected that the driver include the external VLAN header as part of the transmit data structure. The software may post the internal VLAN header as part of the transmit data structure or embedded in the transmit descriptor (see Section 26.2.2 for details). the Controller does not relate to the external VLAN header other than the capability of "skipping" it for parsing of inner fields.

Notes:

- The VLAN header in a packet that carries a single VLAN header is treated as the external VLAN.
- The Controller expects that any transmitted packet has at least the external VLAN added by the software. For those packets where an external VLAN is not present, any offload that relates to inner fields to the EtherType may not be provided.

26.4.5.2 Receive Behavior With External VLAN

When a port of the Controller is working in this mode, the Controller assumes that all packets received by this port have at least one VLAN, including packet received or sent on the manageability interface.

One exception to this rule are flow control PAUSE packets which are not expected to have any VLAN. Other packets may contain no VLAN, however a received packet that does not contain the first VLAN is forwarded to the host but filtering and offloads are not applied to this packet.

See the table below for the supported receive processing when the device is set to "Double VLAN" mode.

Stripping of VLAN is done on the second VLAN if it exists. All the filtering functions of the Controller ignore the first VLAN in this mode.

The presence of a first VLAN tag is indicated it in the *RDESC.STATUS.VEXT* bit.

Queue assignment of the Rx packets is not affected by the extended VLAN header. It may depend on the internal VLAN, MAC address or any upper layer content as described in Section 26.1.1.

Table 26-53. Receive Processing in Double VLAN Mode

VLAN Headers	Status.VEXT	Status.VP	Packet Parsing	Rx offload functions
Extended and internal	1	1	+	+
Internal Only	Not supported			
V-Ext	1	0	+	+
None[1]	0	0	+ (flow control only)	-

1. A few examples for packets that may not carry any VLAN header may be: Flow control and Priority Flow Control; LACP; LLDP; GMRP; 802.1x packets

26.5 Configurable LED Outputs

The Controller implements 4 output drivers (one per port) intended for driving external LED circuits per port. Each LAN device provides an independent LED output - the pin and its function are bound to a specific LAN device.

The configuration for LED outputs is specified via the LEDCTL register. Furthermore, the hardware-default configuration for all the LED outputs, can be specified via EEPROM fields, thereby supporting LED displays configurable to a particular OEM preference.

LED's can be configured to use one of a variety of sources for output indication. The MODE bits control the LED source as described in Table 26-54.

The IVRT bits allow the LED source to be inverted before being output or observed by the blink-control logic. LED outputs are assumed to normally be connected to the negative side (cathode) of an external LED.

The BLINK bits control whether the LED should be blinked (on for 200ms, then off for 200ms) while the LED source is asserted. The blink control might be especially useful for ensuring that certain events which are sufficiently visible by a human eye.

Note: When LED Blink mode is enabled the appropriate LED Invert bit should be set to 0b.

26.5.1 MODE Encoding for LED Outputs

Table 26-54 lists the MODE encoding for LED outputs used to select the desired LED signal source for each LED output.

Table 26-54. Mode Encoding for LED Outputs [1]

Mode	Selected Mode	Source Indication
0010b	LINK_UP	Asserted when any speed link is established and maintained.
0011b	FILTER_ACTIVITY	Asserted when link is established and packets are being transmitted or received that passed MAC filtering.
0100b	LINK/ACTIVITY	Asserted when link is established and when there is no transmit or receive activity.
1001b	FULL_DUPLEX	Asserted when the link is configured for full duplex operation (de-asserted in half-duplex).
1010b	COLLISION	Asserted when a collision is observed.
1011b	ACTIVITY	Asserted when link is established and packets are being transmitted or received.
1101b	PAUSED	Asserted when the PCH’s transmitter is flow controlled.
1110b	LED_ON	Always high (Asserted)
1111b	LED_OFF	Always low (De-asserted)

1. Encoding not shown are reserved.

26.6 Error Correction and Detection

The Controller's main internal memories are protected by error correcting code or parity bits. The larger memories or critical memories are protected by an error correcting code (ECC). The smaller memories are protected either with an error correcting code (ECC for critical memories) or by parity.

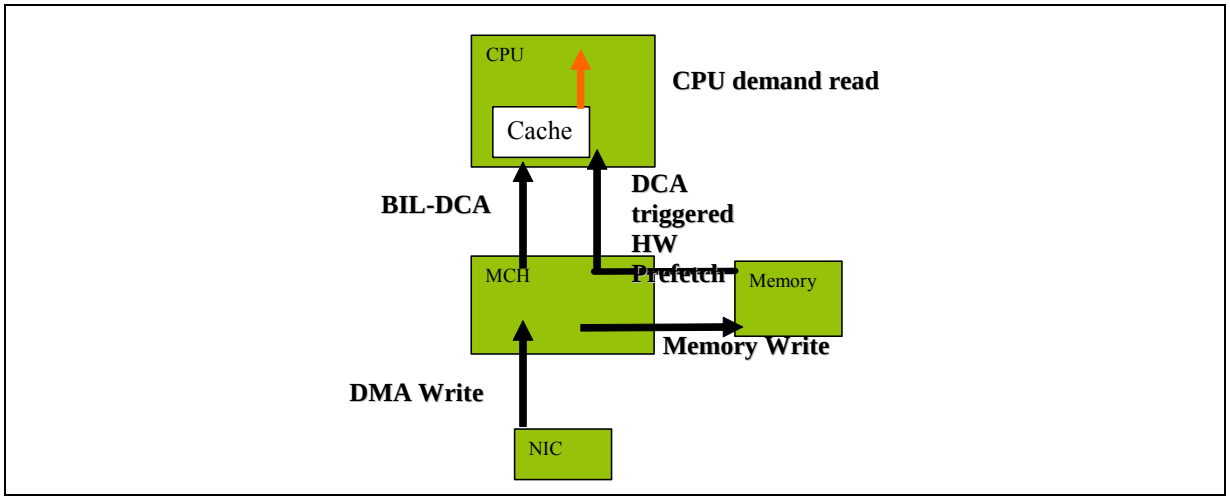
26.7 CPU Affinity Features

26.7.1 Direct Cache Access (DCA)

26.7.1.1 DCA Description

Direct Cache Access (DCA) is a method to improve network I/O performance by placing some posted inbound writes indirectly within CPU cache. DCA requires that memory writes go to host memory and then the processor prefetch the cache lines specified by the memory write. Through research and experiments, DCA has been shown to reduce CPU Cache miss rates significantly.

Figure 26-26.Diagram of DCA Implementation on FSB System



As shown in Figure 26-26, DCA provides a mechanism where the posted write data from an I/O device, such as an Ethernet NIC, can be placed into CPU cache with a hardware pre-fetch. This mechanism is initialized upon a power good reset. A software device driver for the I/O device configures the I/O device for DCA and sets up the appropriate DCA target ID for the device to send data.

DCA implementation is controlled by separated registers (RXCTL and TXCTL) for each receive and transmit queue. In addition, a *DCA Enable* bit can be found in the DCA_CTRL register, and a DCA_ID register can be found for each port, in order to make visible the function, device, and bus numbers to the driver.

The RXCTL and TXCTL registers can be written by software on the fly and can be changed at any time. When software changes the register contents, hardware applies changes only after all the previous packets in progress for DCA have been completed.

There are 2 modes for DCA implementation:

1. Legacy DCA: The DCA target ID is derived from CPU ID.
2. DCA: The DCA target ID is derived from APIC ID.

The software driver selects one of these modes through the DCA_mode register.

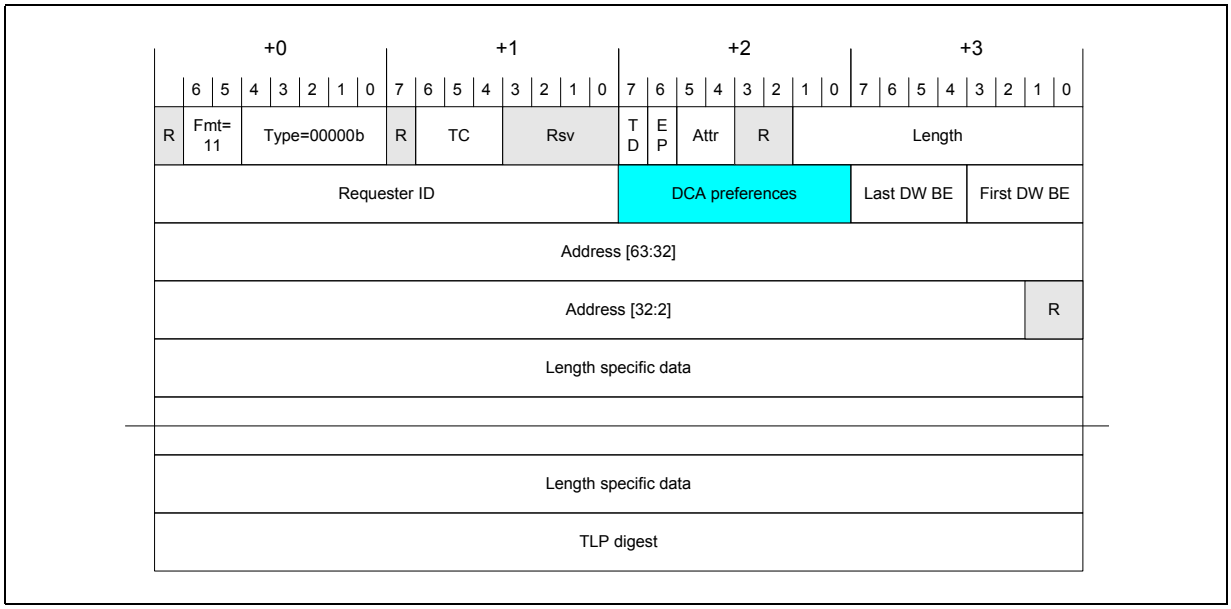
The details of both modes are described below.

26.7.1.2 Details of Implementation

26.7.1.2.1 PCIe* Message Format for DCA

Figure 26-27 shows the format of the PCIe* message for DCA.

Figure 26-27. PCIe* Message Format for DCA



The DCA preferences field has the following formats.

Table 26-55. Legacy DCA Systems

Bits	Name	Description
0	DCA indication	0b: DCA disabled 1b: DCA enabled
4:1	DCA Target ID	The DCA Target ID specifies the target cache for the data.
7:5	Reserved	Reserved

Table 26-56. DCA Systems

Bits	Name	Description
7:0	DCA target ID	0000.0000b: DCA is disabled Other: Target Core Id derived from APIC Id. The method for this is described in DCA Platform Architecture Specification, section 7.3.1

Note: All functions within the Controller have to adhere to the “tag encoding” rules for DCA writes. Even if a given function is not capable of DCA, but other functions are capable of DCA, memory writes from the non-DCA function must set the *Tag* field to “00000000”.

26.8 Virtualization

I/O virtualization is a mechanism to share I/O resources among several consumers. For example, in a virtual system, multiple operating systems are loaded and each executes as though the whole system's resources were at its disposal. However, for the limited number of I/O devices, this presents a problem because each operating system might be in a separate memory domain and all the data movement and device management has to be done by a VMM (Virtual Machine Monitor). VMM access adds latency and delay to I/O accesses and degrades I/O performance. Virtualized devices are designed to reduce the burden of VMM by making certain functions of an I/O device shared and thus can be accessed directly from each guest operating system or Virtual Machine (VM).

Two modes to support operation in a Virtualized environment were implemented in previous products:

1. Direct assignment of part of the port resources to different guest OSes using the PCI sig SR-IOV standard. Also known as “Native mode” or pass through mode. This mode is referenced as IOV mode through this chapter
2. Central management of the networking resources by an IOVM or by the VMM. This mode is referenced as VMDq mode. Two VMDq modes exist, VMDq1 and a more advanced version named VMDq2.

In a virtualized environment, the Controller serves up to 8 virtual machines (VMs).

The Controller supports fully VMDq1 mode and partially VMDq2 mode and does not support SR-IOV. In a virtualized environment, the Controller serves up to 8 virtual machines (VMs) per port. The Controller’s 16 queues can be accessed by 8 different VMs if configured properly. When the Controller is enabled for multiple queue direct access for VMs, it becomes a VMDq device.

Note: Most configuration and resources are shared across queues. System software must resolve any conflicts in configuration between the VMs.

The following section describes the support the Controller provides for VMDq.

26.8.1 Assignment of MSI-X Vectors to VM

MSI-X vectors are used for three purposes:

1. Differentiation of interrupt causes that avoids the need to read an interrupt cause register.
2. Assignment of different interrupt handling to different CPUs.
3. The implementation of interrupts in the Controller adds another use of allowing different interrupt moderation rates.

MSI-X vectors are allocated for cause differentiation and interrupt rate differentiation. The interrupts that the VM driver needs to handle are Tx packet sent per queue, Rx packet received per queue, TCP timer and some special events.

26.8.2 VM Resource Summary

The Controller supports 8 VMs per port, each VM can utilize 2 queue pair (Tx/Rx) per VM and 3 MSI-X vectors per VM. If the amount of VMs supported is less than 8, the available resources can be distributed between the active VMs.

26.8.3 Packet Switching (VMDq) Model

26.8.3.1 VMDq Assumptions

To support receive packet replication it is assumed that Broadcast and Multicast traffic volume is relatively low. In the case of large Broadcast and Multicast traffic volume that causes congestion on the PCIe interface, the Storm Control circuitry can be programmed to drop Broadcast and Multicast packets or flow control can be activated to avoid internal receive buffer overflow.

26.8.3.2 VM Selection

The VM selection is done by MAC address and VLAN tag. Broadcast and Multicast packets are forwarded according to the individual setting of each VM and might be replicated to multiple VMs.

26.8.3.2.1 Filtering Capabilities

The following capabilities exists in to decide what is the final destination of each packet in addition to the regular L2 filtering capabilities:

- MAC addresses filters (RAH/RAL registers) for both unicast and multicast filtering. These are shared with L2 filtering. For example, the same MAC addresses are used to determine if a packet is received by the switch and to determine the forwarding destination.
- Shared VLAN filters (VLVF registers) - each VM can be made member of each VLAN.
- Multicast exact filtering using the existing remaining RAH/RAL registers otherwise an imperfect multicast table is shared between VMs.
- 256 hash filtering of multicast addresses shared between the VMs (MTA table).
- Promiscuous unicast, multicast & enable broadcast per VM.

Note: Packets for which no queueing decision was done and still accepted by the L2 filtering, is directed to the queue pool of the default VM or dropped.

26.8.3.3 L2 Filtering

L2 filtering is the 1st stage of 3 stages that determine the destination of a received packet. The 3 stages are defined in Section 26.1.1.

All received packets pass the same filtering as in the non virtualized case; regular VLAN filtering using the global VLAN table (*VFTA*) and filtering according to the *RAH/RAL* registers and according to the various promiscuous bits.

Note: Every VLAN tag set in the *VLVF* registers should be asserted also in the *VFTA* table.

- The *RCTL.UPE* bit (Promiscuous unicast) is not available per VM and might be modified by the IOVM or VMM.

26.8.3.4 VMDq Receive Packets Switching

Receive packet switching is the 2nd stage of 3 stages that determine the destination of a received packet. The 3 stages are defined in Section 26.1.1.

In this stage the VM is identified by the "pool list" as described in Section 26.8.3.4.1 and Section 26.8.3.4.2.

When working in a virtualized environment, a single receive queue can still be determined by the Ethertype filters. If these filters don't match, then a pool list should be found.

When working in replication mode, broadcast and multicast packets can be forwarded to more than one VM, and can be replicated to more than one receive queue. Replication is enabled by the *Rpl_En* bit in the VT_CTL register.

In virtualization mode, the pool list is a list of one or more VMs to which the packet should be forwarded. The pool list is used in choosing the target queue list except for cases in which high priority filters take precedence. There is a difference in the way the pool list is generated when replication mode is enabled or disabled.

26.8.3.4.1 VMDq Replication Mode Enabled

When replication mode is enabled (VT_CTL.*Rpl_En* = 1), each broadcast/multicast packet can go to more than one pool. Finding the pool list should be done according to the following steps:

1. Exact unicast or multicast match - If there is a match in one of the exact filters (RAL/RAH), for unicast or multicast packets, take the *RAH.POOLSEL[7:0]* field as a candidate for the pool list.
2. Broadcast - If the packet is a broadcast packet, add pools for which their *VMOLR.BAM* bit (Broadcast Accept Mode) is set.
3. Unicast hash - If the packet is a unicast packet, and the prior steps yielded no pools, check it against the Unicast hash table (UTA). If there is a match, add pools for which their *VMOLR.ROPE* bit (Receive Overflow packet enable) is set.
4. Multicast hash - If the packet is a multicast packet and the prior steps yielded no pools, check it against the multicast hash table (MTA). If there is a match, add pools for which their *VMOLR.ROMPE* bit (Receive Multicast packet enable) is set.
5. Multicast Promiscuous - If the packet is a multicast packet, take the candidate list from prior steps and add pools for which their *VMOLR.MPE* bit (Multicast Promiscuous Enable) is set.
6. Ignore MAC (VLAN only filtering) - If *VT_CTL.IGMAC* bit is set, then the previous steps are ignored and a full pool list is assumed for the next step.
7. VLAN groups - This step is relevant only if the *RCTL.VFE* bit is set, otherwise it is skipped. Packets should be sent only to VMs that belong to the packet's VLAN group.
   a. Tagged packets: enable only pools in the packet's VLAN group as defined by the VLAN filters - *VLVF[n].VLAN_id* and their pool list - *VLVF[n].POOLSEL[7:0]*.
   b. Untagged packets: enable only pools with their VMOLR.AUPE bit set
   c. If there is no match, the pool list should be empty.

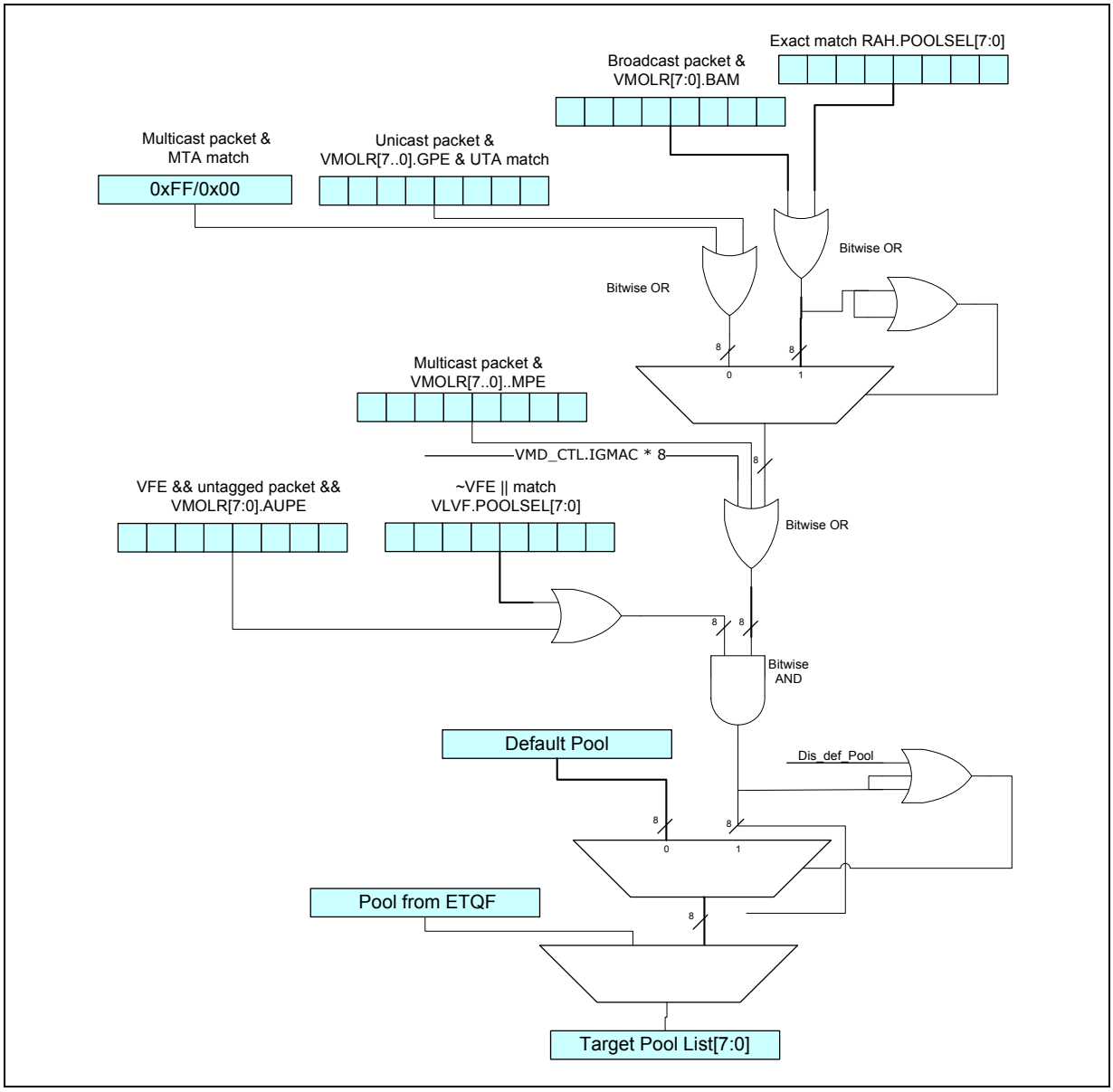
Note: In a VLAN network, untagged packets are not expected. Such packets received by the switch should be dropped, unless their destination is a virtual port set to receive these packets. The setting is done through the *VMOLR.AUPE* bit. It is assumed that VMs for which this bit is set are members of a default VLAN and thus only MAC queuing is done on these packets.

8. Default Pool - If the pool list is empty at this stage and the *VT_CTL.Dis_Def_Pool* bit is not set, then set the default pool bit in the target pool list (from VT_CTL.Def_PL).

9. Ethertype filters - If the one of the Ethertype filters (ETQF) matches the packet and queuing action is requested, the VM list is set to the pool pointed by the filter.
10. Mirroring - For each of the 4 mirroring rules add the destination (mirroring) pool (VMRCTL.MP) to the pool list according to the following rules:
    a. Pool mirroring - if VMRCTL.VPME is set and one of the bits in the pool list matches one of the bits in the VMRVM register.
    b. VLAN port mirroring - if VMRCTL.VLME is set and the index of the VLAN of the packet in the VLVF table matches one of the bits in the VMRVLAN register.
    c. Uplink port mirroring - if VMRCTL.UPME is set and the pool list is not empty and the packet came from the LAN.
11. Length Limit: If the packet is longer than a legal Ethernet packet, remove from the pool list all the pools for which the *VMOLR.LPE* bit is not set or for which the packet length is larger than the value in the *VMOLR.RLPML* field.

The above process, up to stage 9. can be logically described by the following scheme:

Figure 26-28.Pool List Selection - Replication Enabled



26.8.3.4.2 VMDq Replication Mode Disabled

When replication mode is disabled (VT_CTL.*Rpl_En* = 0), the software should take care of multicast and broadcast packets and check which of the VMs should get them. In this mode the pool list always contains one pool only according to the following steps:

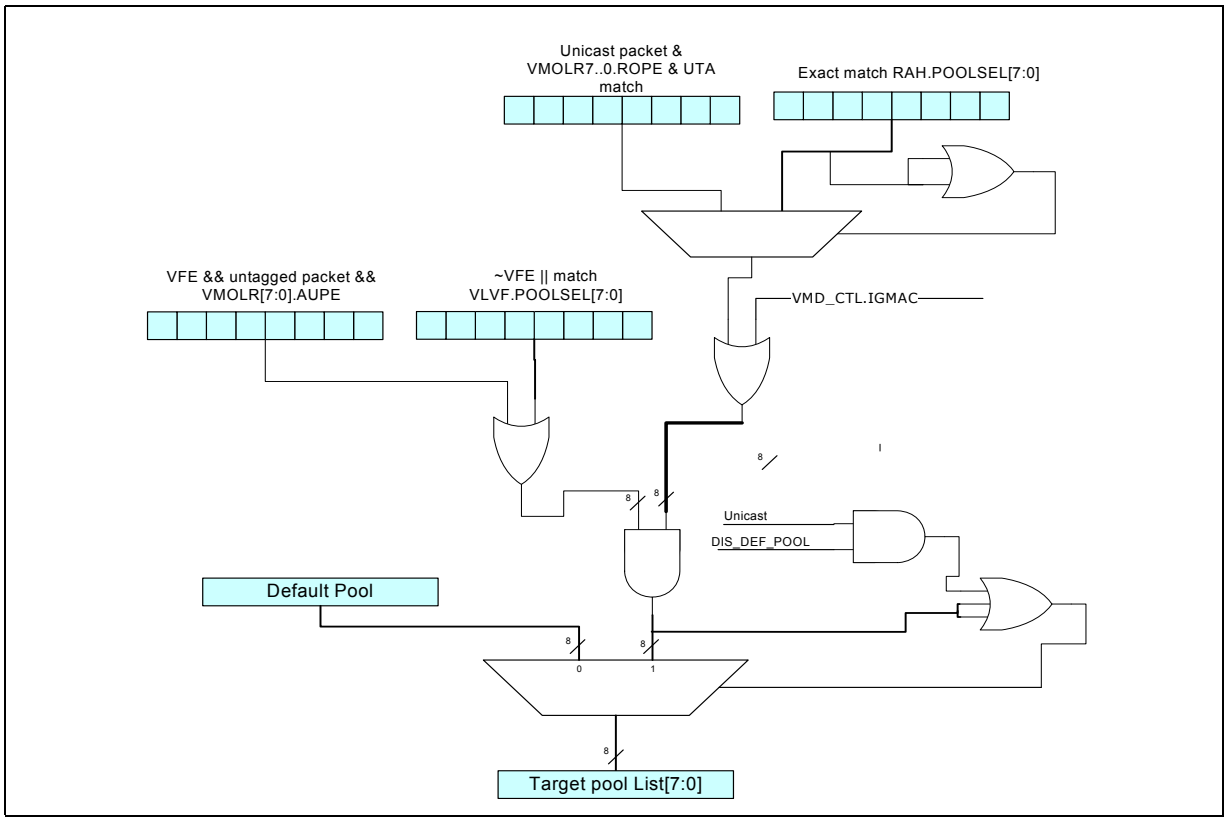
1. Exact unicast or multicast match - If the packet DA matches one of the exact filters (RAL/RAH), take the *RAH.POOLSEL[7:0]* field as a candidate for the pool list.
2. Ignore MAC (VLAN only filtering) - If *VT_CTL.IGMAC* bit is set, then the previous step is ignored and a full pool list is assumed for the next step.
3. Unicast hash - If the packet is a unicast packet, and the prior steps yielded no pools, check it against the Unicast hash table (UTA). If there is a match, add the

pool for which the *VMOLR.ROPE* bit (Receive Overflow packet enable) is set. (See software limitation no 3. below).

4. VLAN groups - This step is relevant only if the *RCTL.VFE* bit is set, otherwise it is skipped. Packets should be sent only to VMs that belong to the packet's VLAN group.
   a. Tagged packets: enable only pools in the packet's VLAN group as defined by the VLAN filters - *VLVF[n].VLAN_id* and their pool list - *VLVF[n].POOLSEL[7:0]*.
   b. Untagged packets: enable only pools with their VMOLR.AUPE bit set
   c. If there is no match, the pool list should be empty.
5. Default pool- If the packet is a unicast packet and no pool was chosen and the *VT_CTL.Dis_Def_Pool* bit is not set, then set the default pool bit in the pool list (from VT_CTL.Def_PL).
6. Broadcast or Multicast - If the packet is a Multicast or Broadcast packet and was not forwarded in step 1 and 2, set the default pool bit in the pool list (from VT_CTL.Def_PL).
7. Length Limit: If the packet is longer than a legal Ethernet packet, remove from the pool list all the pools for which the *VMOLR.LPE* bit is not set or for which the packet length is larger than the value in the *VMOLR.RLPML* field.

The above process can be logically described by the following scheme:

Figure 26-29.Pool List Selection - Replication Disabled



The following limitations applies when replication is disabled:

1. It is the software responsibility to not set more than one bit in the bitmaps of the exact filters. Multiple bits might be set in an RAH register as long as it is guaranteed that the packet is sent to only one queue by other means (VLAN)
2. The software must not set per-VM promiscuous bits (multicast or broadcast).
3. The software must not set the *Rope* bit in more than one *VMOLR* register.
4. If *VT_CTL.IGMAC* bit is set, the software must not set the *VMOLR.AUPE* in more than one VMOLR register and must not set more than one bit in each of the *VLVF.POOLSEL* bitmaps.
5. The software must not activate mirroring.
6. The software should take care not to set the *Rope* bit in more than one *VMOLR* register.

26.8.3.5 Mirroring Support

The Controller supports 4 mirroring rules. Each rule can be of one of 3 types. Only Egress mirroring is supported and not ingress. For example, the mirroring is done on the receive path and mirrored packets reflect all the changes that occur to the received packet (for example, Vlan stripping). Mirroring is supported only to virtual ports and not to the uplink (for example, a mirrored packet can not be sent back to the Network).

Mirroring should be activated only when one of the VMDq queueing modes is used.

The following types of rules are supported:

1. Virtual port mirroring - reflects all the packets sent to a set of given VMs.
2. Uplink port mirroring - reflects all the traffic received from the network.
3. VLAN mirroring - reflects all the traffic received in a set of given VLANs.

All the modes can be accumulated into a single rule.

This new mirroring mode is controlled by a set of rule control registers:

- *VMRCTL* - controls the rules to be applied and the destination port.
- *VMRVLAN* - controls the VLAN ports as listed in the *VLVF* table taking part in the VLAN mirror rule.
- *VMRVM* - controls the VMs ports taking part of the Virtual port mirror rule.

Mirroring is supported only when replication is enabled. The exact flow of mirroring is described in step 10. in Section 26.8.3.4.1.

26.8.3.6 VMDq Offload Support

In case of packets directed to one VM only, the offloads are determined by this specific VM setting. However, the Controller can not apply different offloads (VLAN & CRC strip + decision of size of header for split/replication offload) to different replication of the same packet. The following sections describes the rules used to decide which offloads to apply in case of replicated packets.

If replication is disabled, the offloads are determined by the unique destination of the packet.

Note: In a virtualization environment (*MRQC.Multiple Receive Queues Enable* = 011b), the global VLAN strip and CRC strip bits (*CTRL.VME* & *RCTL.SECRC*) are ignored and the VM specific bits in VMOLR & RPLOLR are used instead.

26.8.3.6.1 Replication by Exact MAC Address

As mentioned above, the same MAC address can be assigned to more than one VM. This is used for the following cases:

- Multicast address - In this case, the different VMs might be part of the same VLAN. The offloads applied to packets matching this address are defined in the replicated packets offloads register (RPLOLR register).
- Unicast - Same MAC different VLAN - In this case, each VM should belong to different VLAN(s). The applied offloads is according to the pool selected by the MAC/VLAN pair. One exception is the VLAN strip decision which is done according to the first pool parameters. This means that all the pools sharing a MAC address should use a common VLAN strip policy.

26.8.3.6.2 Replication by Promiscuous Modes

A packet might be replicated to multiple VMs because part of the VMs are set to receive all multicast or broadcast packets or because of a packet matching one of the hash tables (UTA or MTA).

The offloads applied to packet are defined in the replicated packets offloads registers (RPLOLR register).

In case of unicast packet, the offloads are applied according to the first pool selected to receive the packet.

26.8.3.6.3 Replication by Mirroring

Offloads of mirrored packets are determined according to the original pool.

26.8.3.6.4 VLAN Only Filtering

If *VT_CTL.IGMAC* bit is set, the pool is defined according to the VLAN only. In this mode, only a uniform VLAN strip policy is supported. This means that the *VMOLR.STRVLAN* bit should be set to the same value for all VMs.

26.8.3.6.5 Small Packets Padding

In Virtualized systems, the driver receiving the packet in the VM might not be aware of all the hardware offloads applied to the packet. Thus, in case of stripping actions by the hardware (VLAN strip), it might receive packets which are smaller than a legal packet. The Controller provides an option to pad small packets in such cases so that all packets have a legal size. This option can be enabled only if the CRC is stripped. In these cases, all packets are padded to 60 bytes (legal packet - 4 bytes CRC). The padding is done with zero data. This function is enabled via the *RCTL.PSP* bit.

26.8.3.7 Security Features

The Controller allows some security checks on the inbound and outbound traffic of the switch.

26.8.3.7.1 Inbound Security

Each incoming packet from the LAN is filtered according to the VLAN tag so that packets from one VLAN can not be received by VMs that are not members of that VLAN.

26.8.3.7.2 Interrupt on Misbehavior of VM (Malicious Driver Detection)

The hardware can be programmed to take some action as a result of some misbehavior of a VM. These actions might hint to the fact that some VM is malicious and the VMM should remedy the situation. In order to inform the VMM of this fact, an interrupt bit exists in the ICR register (*ICR.MDDET* bit) to indicate the occurrence of such behavior. The LVMMC register contains information on which queue (*LVMMC.Last_Q*) and port (*LVMMC.Mal_PF*) the malicious behavior was detected. The *LVMMC* register is clear by read.

Malicious driver behavior detection is enabled by setting the *DTXCTL.MDP_EN* bit to 1. In addition to fully enable the Malicious driver detection functionality the DTXCTL.OutOfSyncEnable bit should also be set to 1b. On detection of a malicious driver event the Controller stops activity of the offending queue and generates an interrupt by asserting the *ICR.MDDET* bit. Cause of Malicious driver activation is reported in the *LVMMC* register. To re-activate offending queue driver should write a 0 to the relevant bit in the *MDFB* register.

26.8.3.7.3 Storm Control

As there is no separate path for multicast & broadcast packets, too much replicated packets might cause congestions in the data path. In order to avoid such scenarios, broadcast and multicast storm control rate limiters are added. The rate controllers define windows and the maximal allowed number of multicast or broadcast bytes/packets per window. Once the threshold is crossed different types of policies can be applied.

26.8.3.7.3.30 Assumptions

- Only one interval size and interval counter is used for both broadcast & multicast storm control mechanisms.
- The threshold and actions for each mechanism are separate.
- The traffic used to calculate the broadcast & multicast rate is all the traffic with a local destination.
- The storm control does not block traffic to the network.
- The basic unit of traffic counted is 64 bytes of data.

26.8.3.7.3.31 Storm Control Functionality

The time interval over which Broadcast Storm control is performed is controlled by three factors.

- SCBI register
- Port speed.
- The value in SCCRL.INTERVAL

The first two factors determine the Unit time interval as described in Table 26-57. The interval is automatically chosen internal to hardware based on port speed. The third factor (*Interval* field) determines how many of such unit intervals are considered for one Storm Control Interval.

Table 26-57. Storm Control Interval by Speed

Port Speed	MIN Time Interval	MAX Time Interval
1 Gb/s	100 µs	100 ms
100 Mb/s	1 ms	1 s
10 Mb/s	10 ms	10 s

The number of 64 bytes chunk of Broadcast or Multicast packets that are allowed in a given interval is determined by setting the BSCTRH or MSCTRH register respectively.

The Controller supports two modes of reactions to storm event:

1. Block all Multicast or Broadcast packets from the moment the threshold is crossed until the end of the interval. The block is removed at the end of the interval until the threshold is crossed again. This mode is set by asserting SCCRL.MDICW (for multicast) or SCCRL.BDICW (for broadcasts). This mode is used as a rate limiter.
2. Block all Multicast or Broadcast packets from the moment the threshold is crossed until a full interval without threshold crossing is registered. The block is removed at the end of the interval until the threshold is crossed again. This mode is set by asserting SCCRL.MDICW and SCCRL.MDIPW (for multicast) or SCCRL.BDICW and SCCRL.BDIPW (for broadcasts). This mode is used for storm blocking.

The Controller can be programmed to add all packets for which a queue was not found for storm control calculation. For example, packets that passed the 1st stage of L2 filtering but didn't pass the 2nd stage of pooling, or where sent to the default pool, as broadcast packets. This mode is activated by setting the *SCCRL.BIDU* field.

Any change in the storm control state (block or pass of multicast or broadcast packets) is indicated to the software via the ICR.SCE interrupt cause. The current state is reflected in the SCSTS register.

For diagnostic purpose only, the storm control timer and counters can be read via the SCTC, MSCCNT & BSCCNT registers.

26.8.3.8 External Switch Loopback Support

One of the long term solutions for the switching issue is a mode where an external switch would do the loopback of VM to VM traffic and the NIC is responsible for the replication of multicast packets only. In order to support this mode the received packets SA should be compared to the exact MAC addresses to check if the packet originated from a local source, so that the packet is not forwarded to the VM originator. This mode is enabled by the *VT_CTL.FLP* bit.

26.8.3.9 Switch Control

The VMM/IOVM driver has some control of the switch logic. The following registers are available to the VMM/IOVM for this purpose:

VLVF: VLAN queuing table: A set of 32 VLAN entries with an associated per VM bit map allowing allocation of each VM to each of the 32 VLAN tags.

VT_CTL: VT Control register - contains the following fields:

- Replication enable - allows replication of multicast & broadcast packets. If this bit is cleared, Rx multicast & broadcast packets are sent to the default VM.
- Default pool - defines where to send packets that passed L2 filtering but didn't pass any of the queueing mechanisms.
- Default pool disable- defines whether to drop packets that passed L2 filtering but didn't pass any of the queueing mechanisms.

VMOLR/RPMOLR: Defines the offloads and pool selection options for each VM and for replicated packets.

In addition the storm control mechanism is programmed as described in Section 26.8.3.7.3.31.

26.8.4 Virtualization of the Hardware

This section describes additional features used in VMDq mode.

26.8.4.1 Per Pool Statistics

Part of the statistics are by definition shared and can not be allocated to a specific VM. For example, CRC error count can not be allocated to a specific VM, as the destination of such a packet is not known if the CRC is wrong.

All the non specific statistics is handled by the VMM in the same way as it is done in non virtualized systems. A VM might require a statistic from the VMM but might not access it directly.

The conceptual model used to gather statistics in a virtualization context is that each queue pool is considered as a virtual link and the Ethernet link is considered as the uplink of the switch. Thus any packet sent by a VM is counted in the Tx statistics, even if it was dropped by the MAC from some reason. In the same way, a replicated packet is counted in each of the VMs receiving it.

The following statistics are be provided per VM:

1. Good Packet received count (*VFGPRC*).
2. Good Packet transmitted count (*VFGPTC*).
3. Good octets received count (*VFGORC*).
4. Good octets transmitted count (*VFGOTC*).
5. Multicast Packets Received Count (*VFMPRC*).

Note: All the per VM statistics are read only (RO) and wrap around after reaching their maximal value.

26.9 Time SYNC (IEEE1588 and IEEE 802.1AS)

IEEE 1588 addresses the clock synchronization requirements of measurement and control systems. The protocol supports system-wide synchronization accuracy in the sub-microsecond range with minimal network and local clock computing resources. The protocol is spatially localized and allows simple systems to be installed and operated without requiring the administrative attention of users.

The IEEE802.1AS standard specifies the protocol used to ensure that synchronization requirements are met for time sensitive applications, such as audio and video, across Bridged and Virtual Bridged Local Area Networks consisting of LAN media where the transmission delays are fixed and symmetrical; for example, IEEE 802.3 full duplex links. This includes the maintenance of synchronized time during normal operation and following addition, removal, or failure of network components and network reconfiguration. It specifies the use of IEEE 1588 specifications where applicable.

Activation of the the Controller Time Sync mechanism is possible in full duplex mode only. No limitations on wire speed exist, although wire speed might affect the accuracy. Time Sync protocol is tolerant of dropping packets as well as missing timestamps.

26.9.1 Flow and Hardware/Software Responsibilities

The operation of a PTP (Precision Time Protocol) enabled network is divided into two stages, initialization and time synchronization.

At the initialization stage every master enabled node starts by sending Sync packets that include the clock parameters of its clock. Upon reception of a Sync packet a node compares the received clock parameters to its own. If the received clock parameters of a peer are better, the node moves to Slave state and stops sending Sync packets. When in slave state the node continuously compares the incoming packet clock parameters to its currently chosen master. If the new clock parameters are better then the current master selection, it changes master clock source. Eventually the best master clock source is chosen. Every node has a defined Sync packet time-out interval. If no Sync packet is received from its chosen master clock source during the interval it moves back to master state and starts sending Sync packets until a new Best Master Clock (BMC) is chosen.

The time synchronization stage is different for master and slave nodes. If a node is in master state it should periodically send a Sync packet which is time stamped by hardware on the transmit path (as close as possible to the PHY). After the Sync packet a Follow_up packet is sent which includes the value of the timestamp kept from the Sync packet. In addition the master should timestamp Delay_Req packets on its RX path and return to the slave that sent it the timestamp value using a Delay_Response packet. A node in Slave state should timestamp every incoming Sync packet that is received from its selected master, software uses this value for time offset calculation. In addition it should periodically send Delay_Req packets in order to calculate the path delay from its master. Every sent Delay_Req packet sent by the slave is time stamped and kept. Using the value received from the master Delay_Response packet the slave can now calculate the path delay from the master to the slave. The synchronization protocol flow and the offset calculation are described in the following figure.

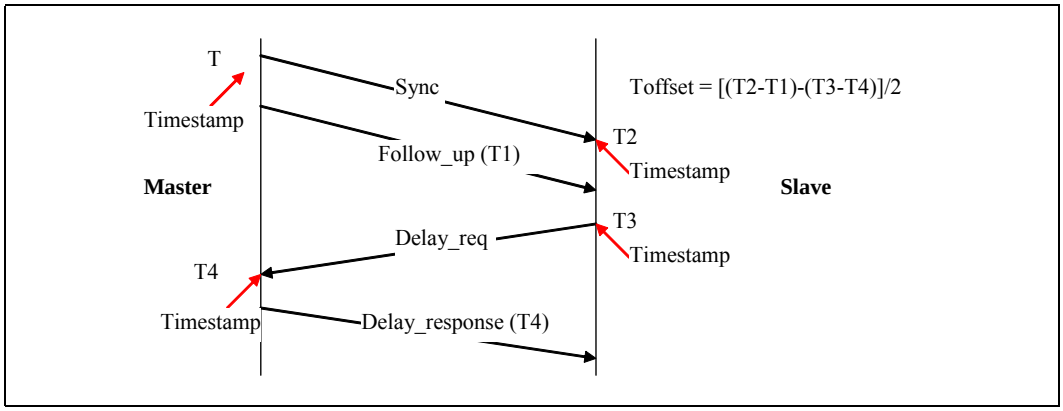


Figure 26-32.Sync Flow and Offset Calculation

The hardware's responsibilities are:

1. Identify the packets that require time stamping.
2. Time stamp the packets on both receive and transmit paths.
3. Store the time stamp value for software.
4. Keep the system time in hardware and give a time adjustment service to the software.
5. Maintain auxiliary features related to the system time.

The software's responsibilities are:

1. BMC protocol execution which means defining the node state (master or slave) and selection of the master clock if in slave state.
2. Generate PTP packets, consume PTP packets.
3. Calculate the time offset and adjust the system time using the hardware mechanism.
4. Enable configuration and usage of the auxiliary features.

Table 26-58. Chronological Order of Events for Sync and Path Delay

Action	Responsibility	Node Role
Generate a Sync packet with timestamp notification in descriptor	Software	Master
Timestamp the packet and store the value in registers (T1)	Hardware	Master
Timestamp incoming Sync packet, store the value in register and store the sourceID and sequenceID in registers (T2)	Hardware	Slave
Read the timestamp from register, prepare a Follow_Up packet and send	Software	Master
Once Follow_Up packet is received, load T2 from registers and T1 from Follow_up packet	Software	Slave
Generate a Delay_Req packet with timestamp notification in descriptor	Software	Slave
Timestamp the packet and store the value in registers (T3)	Hardware	Slave
Timestamp incoming Delay_Req packet, store the value in register and store the sourceID and sequenceID in registers (T4)	Hardware	Master
Read the timestamp from register and send back to Slave using a Delay_Response packet	Software	Master
Once Delay_Response packet is received, calculate offset using T1, T2, T3 and T4 values	Software	Slave

26.9.1.1 TimeSync Indications in Receive and Transmit Packet Descriptors

Certain indications are transferred between software and hardware regarding PTP packets.

On the transmit path the software should set the *1588* bit in the transmit packet descriptor (MAC field bit 1). To indicate that the transmit packet time stamp should be taken and placed in the *TXSTMPH* and *TXSTMPL* time stamp registers.

On the receive path the hardware transfers three indications to software in the receive descriptor:

1. An indication in *RDESC.Packet Type* that this packet is a PTP packet (no matter if timestamp is sampled or not). This indication is used also by PTP packets required for protocol management.

Note: This indication is only relevant for L2 type packets (the PTP packet is identified according to its Ethertype). PTP packets have the *L2Type* bit in the *Packet Type* field set (bit 11) and the Etype matches the filter number set by the software to filter PTP packets. UDP type PTP packets don't require such an indication since the port number (319 for event and 320 for all other PTP packets) directs the packets toward the time sync application.

2. A second indication in the *RDESC.STATUS.TS* bit to indicate to the software that time stamp was taken for this packet and placed in the *RXSTMPH* and *RXSTMPL* time stamp registers. Software needs to access the time stamp registers to get the time stamp values.
3. A third indication in the *RDESC.STATUS.TSIP* bit to indicate that a time stamp was taken for this packet and placed at the start of the receive buffer (For further information see Section 26.1.10).

26.9.2 Hardware Time Sync Elements

All time sync hardware elements are reset to their initial values as defined in the registers section upon MAC reset.

26.9.2.1 System Time Structure and Mode of Operation

The time sync logic contains the SYSTIM counter to maintain the system time value. This is a 72 bit counter that is built of the *SYSTIMR*, *SYSTIML* and *SYSTIMH* registers. When in Master state the *SYSTIMH*, *SYSTIML* and *SYSTIMR* registers should be set once by the software according to general system requirements, when in slave state software should update the system time on every sync event as described in Section 26.9.2.3. Setting the system time is done by direct write to the *SYSTIMH* register and fine tuning the setting of the SYSTIM register, using the adjustment mechanism described in Section 26.9.2.3.

Read access to the *SYSTIMH*, *SYSTIML* and *SYSTIMR* registers should be executed in the following order:

1. Software reads register *SYSTIMR*.

Note: At this stage the hardware latches the value of *SYSTIMH and SYSTIML* registers.

2. Software reads register *SYSTIML*.
3. Software reads register *SYSTIMH.*

Note: The latched *SYSTIMH* and *SYSTIML* values (from last read of *SYSTIMR* register) should be returned by hardware when reading the *SYSTIML* and *SYSTIMH* registers.

The SYSTIM timer value in the *SYSTIMH*, *SYSTIML* and *SYSTIMR* registers, is updated periodically each 8 nS clock cycle according to the following formula:

$$\text{New SYSTIM} = \text{Old SySTIM} + 8\ \text{nS} +/- \textit{TIMINCA.Incvalue} * 2^{-32}\ \text{nS}$$

Where subtraction or addition of the *TIMINCA.Incvalue* value is defined according to the *TIMINCA.ISGN* value (0 - Add, 1 - Subtract). For the *TIMINCA* register description see section 28.16.1.12.

26.9.2.2 Time Stamp Mechanism

The time stamp logic is located on transmit and receive paths at a location as close as possible to the PHY, to reduce delay uncertainties originating from implementation differences. The time stamp logic operation is slightly different on transmit and on receive paths.

The transmit logic decides to timestamp a packet if the transmit timestamp is enabled (*TSYNCTXCTL.EN* = 1) and the time stamp bit in the packet descriptor (*TDESD.MAC.1588* = 1) is set. On the transmit side only the time is captured in the TXSTMPL and TXSTMPH registers.

The receive logic parses the received frame and timestamps the receive packet according to the conditions defined in the following fields:

1. *TSYNCRXCTL.Type* field that defines type of packets to be sampled.
2. TSYNCRXCFG.CTRLT field that defines message type criteria for timestamping V1 type packets when TSYNCRXCTL.Type register field equals 001b.
3. *TSYNCRXCFG.MSGT* field that defines message type criteria for timestamping V2 type packets when *TSYNCRXCTL.Type* register field equals 000b or 010b.

When conditions to timestamp a receive packet are met, the timestamp is latched in the *RXSTMPL and RXSTMPH* registers and the packet's sourceId and sequenceId are latched in the *RXSATRL* and *RXSATRH* timestamp registers.

Three indications are placed in the receive descriptor to support TimeSync operation:

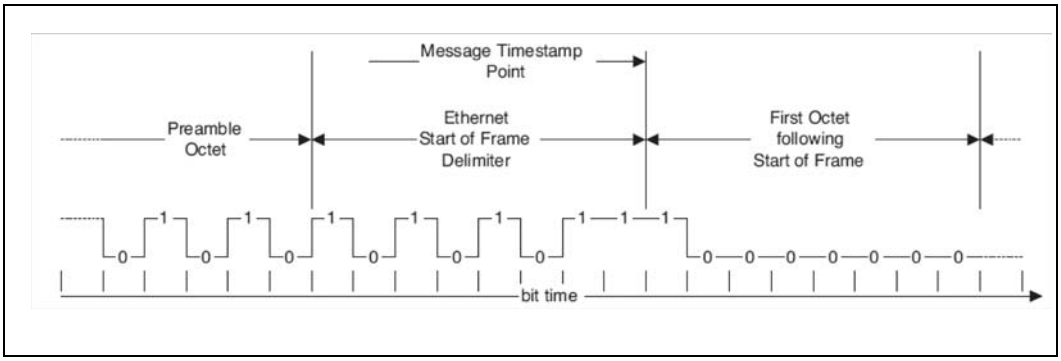
1. *RDESC.Packet Type* - Value in this field identifies that this is a PTP packet (this indication is only for L2 packets since on the UDP packets the port number directs the packet to the application).
2. *RDESC.STATUS.TS* - Bit identifies that a time stamp was taken for this packet and latched in RXSTMPL and RXSTMPH registers and the packet's sourceId and sequenceId are latched in the RXSATRL and RXSATRH timestamp registers.
3. RDESC.STATUS.TSIP - Bit identifies that a time stamp was taken for this packet and placed at the start of the receive buffer (For further information see Section 26.1.10).

For more details see the timestamp registers in Section 28.16. Figure 26-33 defines the exact point where the time value is captured.

On both transmit and receive sides the timestamp values are locked in registers until software reads the TXSTMPH register to unlock Transmit timestamp registers or reads the RXSTMPH register to unlock the receive timestamp registers. As a result, if a new PTP packet that needs to be time stamped arrives before software accesses the timestamp registers, it is not time stamped. In some cases on the receive path a packet that was timestamped might be lost and not reach the host. To avoid a deadlock condition on the time stamp registers the software should keep a watch dog timer to clear locking of the time stamp register. The interval counted by such a timer should be higher then the expected interval between two Sync or Delay_Req packets depends on the node state (Master or Slave).

Note: When *TSYNCRXCTL.Type* value is 100b, the Receive timestamp registers are not locked after a timestamp event.

Figure 26-33.Time Stamp Point



26.9.2.3 Time Adjustment Mode of Operation

A Node in a Time Sync network can be in one of two states Master or Slave. When a Time Sync entity is in the Master state it should synchronize other entities to its System Clock. In this case no time adjustments are needed. When the entity is in slave state it should adjust its system clock by using the data arriving in the Follow_Up and Delay_Response packets and the time stamp values of the Sync and Delay_Req packets. When all the values are available the software in the slave entity can calculate its offset in the following manner:

Toffset = [(T2-T1) - (T3-T4)]/2

T1 - Timing data in Follow_Up packet
T2 - Sync Time Stamp
T3 - Delay_Req Time Stamp
T4 - Timing data in Delay_Response packet

After offset calculation the system time register should be updated. This is done by writing the calculated offset to the *TIMADJL* and *TIMADJH* registers. The order should be as follows:

1. Write the low portion of the offset to the *TIMADJL.TADJL* field.
2. Write the high portion of the offset to the *TIMADJH.TADJH* field with the sign, to define if the offset should be added or subtracted, to the *TIMADJH.Sign* bit.

After the writing to the *TIMADJH* register, the value of *TIMADJH* and *TIMADJL* is added once to the system time (*SYSTIML* and *SYSTIMH*).

26.9.3 Time Sync Related Auxiliary Elements

The time sync logic implements three types of auxiliary elements using the precise system timer (*SYSTIML* and *SYSTIMH*).

26.9.3.1 Target Time

The two target time registers *TRGTTIML/H0* and *TRGTTIML/H1* enable generating a time triggered event to external hardware using one of the SDP pins according to the setup defined in the *TSSDP* and *TSAUXC* registers (See Section 28.16.1.15 and Section 28.16.1.27). Each target time register is structured the same as the system

time register. If the value of the system time is equal or has passed the value written to one of the target time registers, a change in level or a pulse is generated on the programmed SDP outputs.

To generate a level change or pulse on one of the SDP pins driver should:

1. Select SDP pin using the appropriate *TSSDP.TS_SDPx_SEL* field.
2. Program *TRGTTIML/H0* and *TRGTTIML/H1* registers to define time of level change and pulse duration.
3. Program *TSAUXC.PLSGx* bit to define if level change or pulse is driven at Target Time on the selected SDP pin.
   — If pulse generation is selected (*TSAUXC.PLSGx* = 1) program the *TSAUXC.PLSNegx* bit to define if generated pulse is positive or negative.
4. Enable level change or pulse generation by setting the *TSAUXC.EN_TTx* bit.

Each target time register has an enable bit located in the auxiliary control register (*TSAUXC.EN_TTx*). When the SDP level has changed (if *TSAUXC.PLSGx* = 0) or when a pulse is generated (if *TSAUXC.PLSGx* = 1) on the selected SDP pin, the enable bit is cleared and needs to be set again by software to get another target time event.

The Target Time registers (*TRGTTIML/H0* or *TRGTTIML/H1*) can also be used for synchronizing the configurable clock output to the system time (*SYSTIM*). Setting the appropriate *TSAUXC.STx* bit to 1 enables start of clock generation only after Target Time is reached.

26.9.3.2 Configurable Frequency Clock

This feature enables to generate up to 2 programmable clocks on the appropriate SDP pins by configuring the SDP pins using the TSSDP register and by programming appropriate values to the Frequency out Control registers (*FREQOUT0* and *FREQOUT1*). The output clocks are synchronized to the global System (SYSTIM) clock and are affected by System time corrections programmed in the *TIMINCA* register and the *TIMADJL/H* registers.

When clock generation is enabled, the error correction programmed in the *TIMADJL/H* registers is compensated from the clock output gradually, at a rate of 1 ns per 8 nS internal clock cycle. The gradual compensation is done to avoid large duty cycle variations in the output clock. Before updating the *TIMADJL/H* registers, software should verify that the appropriate *TSICR.TADJ0/1* register bit was set to indicate that previous one time adjustment has completed.

To generate either Clock 0 or Clock 1 on one of the SDP pins, the following steps should be taken:

1. Define half period of the clock by programming the *CHCT* field in the relevant *FREQOUT0/1* register. Value placed in *CHCT* field is multiplied by 8 nS to get half clock cycle time.
2. Define SDPx pin that drives clock by programming the appropriate *TSSDP.TS_SDPx_SEL* field and setting the *TSSDP.TS_SDPx_EN* bit to 1 (where x is 0,1,2 or 3).
3. If clock start needs to be aligned to the system time (SYSTIM), program start of clock toggle in the appropriate Target Time (*TRGTTIML0/1* and *TRGTTIMH0/1*) registers and set the relevant *TSAUXC.ST0/1* field to 1.
4. To start clock operation, set the relevant *TSAUXC.EN_CLK0/1* bit to 1.

Clock out drives initially a logical 0. Clock value toggles each time a System Time duration of *FREQOUT0*/1 is reached or passed.

Note: Clock output mechanism should be activated only after SYSTIMH/L timer is aligned to global system clock and SYSTIM timer error correction entered using the *TIMADJL/H* registers is below 64 μs.

26.9.3.3 Time Stamp Events

Upon a change in the input level of one of the SDP pins that was configured to detect Time stamp events using the TSSDP register, a time stamp of the system time is captured into one of the two auxiliary time stamp registers (*AUXSTMPL/H0* or *AUXSTMPL/H1*).

For example to define timestamping of events in the *AUXSTMPL*0 and *AUXSTMPH0* registers, Software should:

1. Set the *TSSDP.AUX0_SDP_SEL* field to select the SDP pin that detects the level change and set the *TSSDP.AUX0_TS_SDP_EN* bit to 1.
2. Set the *TSAUXC.EN_TS0* bit to 1 to enable timestamping.

26.9.4 Time SYNC Interrupts

Time Sync related interrupts can be generated by programming the *TSICR*, *TSIM* and *TSIS* registers. The *TSICR* register logs the interrupt cause, the *TSIM* register enables masking specific *TSICR* bits and the *TSIS* register enables Software generated Time Sync interrupts. Detailed description of the Time Sync interrupt registers can be found in Section 28.16.2. Occurrence of a Time Sync interrupt sets the *ICR.Time_Sync* interrupt bit.

26.9.5 PTP Packet Structure

The time sync implementation supports both the 1588 V1 and V2 PTP frame formats. The V1 structure can come only as UDP payload over IPv4 while the V2 can come over L2 with its Ethertype or as a UDP payload over IPv4 or IPv6. The 802.1AS uses only the layer 2 V2 format.

Table 26-59. V1 and V2 PTP Message Structure

Offset in Bytes	V1 Fields	V2 Fields	
Bits	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	
0	***versionPTP***	transport Specific[1]	***messageType***
1		Reserved	***versionPTP***
2	version Network	message Length	
3			

Table 26-59. V1 and V2 PTP Message Structure (Continued)

Offset in Bytes	V1 Fields	V2 Fields
Bits	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
4	Subdomain	domain Number
5		Reserved
6		flags
7		
8		correctionField
9		
10		
11		
12		
13		
14		
15		
16		reserved
17		
18		
19		
20	message Type	Source Port Identity
21	Source communication technology	
22	Sourceuuid	
23		
24		
25		
26		
27		
28	source port id	
29		
30	***sequenceId***	***sequenceId***
31		
32	***control***	control
33	reserved	Log Message Interval
34	flags	N/A
35		

1. Should be all zero.

Note: Only the fields with the bold italic format colored red are of interest to the hardware.

Table 26-60. PTP Message Over Layer 2

Ethernet (L2)	VLAN (Optional)	PTP Ethertype	PTP message

Table 26-61. PTP Message Over Layer 4

Ethernet (L2)	IP (L3)	UDP	PTP message

When a PTP packet is recognized (by Ethertype or UDP port address) on the receive side then if the version is V1 then the *Control* field at offset 32 should be compared to the *TSYNCRXCFG.CTRLT* message field (see Section 28.16.1.26) otherwise the byte at offset zero should be used for comparison to the *TSYNCRXCFG*.MSGT field. The rest of the required fields are at the same location and size for both V1 and V2.

Table 26-62. Message Decoding for V1 (Control Field at Offset 32)

Enumeration	Value
PTP_SYNC_MESSAGE	0
PTP_DELAY_REQ_MESSAGE	1
PTP_FOLLOWUP_MESSAGE	2
PTP_DELAY_RESP_MESSAGE	3
PTP_MANAGEMENT_MESSAGE	4
reserved	5–255

Table 26-63. Message Decoding for V2 (Message Id Field at Offset 0)

MessageId	Message Type	Value (hex)
PTP_SYNC_MESSAGE	Event	0
PTP_DELAY_REQ_MESSAGE	Event	1
PTP_PATH_DELAY_REQ_MESSAGE	Event	2
PTP_PATH_DELAY_RESP_MESSAGE	Event	3
Unused	Event	4-7
PTP_FOLLOWUP_MESSAGE	General	8
PTP_DELAY_RESP_MESSAGE	General	9
PTP_PATH_DELAY_FOLLOWUP_MESSAGE	General	A
PTP_ANNOUNCE_MESSAGE	General	B
PTP_SIGNALLING_MESSAGE	General	C
PTP_MANAGEMENT_MESSAGE	General	D
Unused	General	E-F

If V2 mode is configured in the TSYNCRXCTL.Type field (see Section 28.16.1.1) then the time stamp should be taken on PTP_PATH_DELAY_REQ_MESSAGE and PTP_PATH_DELAY_RESP_MESSAGE according to the value in the *TSYNCRXCFG*.MSGT message field described in Section 28.16.1.26.

26.10 Statistic Counters

The Controller supports different statistic counters as described in Section 28.18. The statistic counters can be used to create statistic reports as required by different standards. The Controller statistic counters allow support for the following standards:

- IEEE 802.3 clause 30 management – DTE section.
- NDIS 6.0 OID_GEN_STATISTICS.
- RFC 2819 – RMON Ethernet statistics group.
- Linux Kernel (version 2.6) net_device_stats

The following section describes the match between the internal GBE Controller statistic counters and the counters requested by the different standards.

26.10.1 IEEE 802.3 Clause 30 Management

The Controller supports the Basic and Mandatory Packages defined in clause 30 of the IEEE 802.3 spec. The following table describes the matching between the internal statistics and the counters requested by these packages.

Table 26-64. IEEE 802.3 Mandatory Package Statistics

Mandatory package capability	GBE Controller counter	Notes and limitations
FramesTransmittedOK	GPTC	The GBE Controller doesn't include flow control packets.
SingleCollisionFrames	SCC	
MultipleCollisionFrames	MCC	
FramesReceivedOK	GPRC	The GBE Controller doesn't include flow control packets.
FrameCheckSequenceErrors	CRCERRS	
AlignmentErrors	ALGNERRC	

In addition, part of the recommended package is also implemented as described in the following table

Table 26-65. IEEE 802.3 Recommended Package Statistics

Recommended package capability	GBE Controller counter	Notes and limitations
OctetsTransmittedOK	GOTCH/GOTCL	The Controller counts also the DA/SA/LT/CRC as part of the octets. The Controller doesn't count Flow control packets.
FramesWithDeferredXmissions	DC	
LateCollisions	LATECOL	
FramesAbortedDueToXSColls	ECOL	
FramesLostDueToIntMACXmitError	HTDMPC	The Controller counts the excessive collisions in this counter, while 802.3 increments no other counters, while this counter is incremented
CarrierSenseErrors	TNCRS	The Controller doesn't count cases of CRS de-assertion in the middle of the packet.
OctetsReceivedOK	TORL+TORH	The Controller counts also the DA/SA/LT/CRC as part of the octets. Doesn't count Flow control packets.
FramesLostDueToIntMACRcvError	RNBC	
SQETestErrors	N/A	
MACControlFramesTransmitted	N/A	

Table 26-65. IEEE 802.3 Recommended Package Statistics (Continued)

MACControlFramesReceived	N/A	
UnsupportedOpcodesReceived	FCURC	
PAUSEMACCtrlFramesTransmitted	XONTXC + XOFFTXC	
PAUSEMACCtrlFramesReceived	XONRXC + XOFFRXC	

Part of the optional package is also implemented as described in the following table.

Table 26-66. IEEE 802.3 Optional Package Statistics

Optional package capability	GBE Controller counter	Notes
MulticastFramesXmittedOK	MPTC	The Controller doesn't count FC packets
BroadcastFramesXmittedOK	BPTC	
MulticastFramesReceivedOK	MPRC	The Controller doesn't count FC packets
BroadcastFramesReceivedOK	BPRC	
InRangeLengthErrors	LENERRS	
OutOfRangeLengthField	N/A	Packets parsed as Ethernet II packets
FrameTooLongErrors	ROC + RJC	

26.10.2 OID_GEN_STATISTICS

The Controller supports the part of the OID_GEN_STATISTICS as defined by Microsoft* NDIS 6.0 spec. The following table describes the matching between the internal statistics and the counters requested by this structure.

Table 26-67. Microsoft* OID_GEN_STATISTICS (Sheet 1 of 2)

OID entry	GbE Controller counters	Notes
ifInDiscards;	CRCERRS + RLEC + RXERRC + MPC + RNBC + ALGNERRC	
ifInErrors;	CRCERRS + RLEC + RXERRC + ALGNERRC	
ifHCInOctets;	GORCL/GOTCL	
ifHCInUcastPkts;	GPRC - MPRC - BPRC	
ifHCInMulticastPkts;	MPRC	
ifHCInBroadcastPkts;	BPRC	
ifHCOutOctets;	GOTCL/GOTCH	
ifHCOutUcastPkts;	GPTC - MPTC - BPTC	
ifHCOutMulticastPkts;	MPTC	
ifHCOutBroadcastPkts;	BPTC	
ifOutErrors;	ECOL + LATECOL	
ifOutDiscards;	ECOL	
ifHCInUcastOctets;	N/A	
ifHCInMulticastOctets;	N/A	
ifHCInBroadcastOctets;	N/A	

Table 26-67. Microsoft* OID_GEN_STATISTICS (Sheet 2 of 2)

ifHCOutUcastOctets;	N/A	
ifHCOutMulticastOctets;	N/A	
ifHCOutBroadcastOctets;	N/A	

26.10.3 RMON

The Controller supports the part of the RMON Ethernet statistics group as defined by IETF RFC 2819. The following table describes the matching between the internal statistics and the counters requested by this group.

Table 26-68. RMON Statistics

RMON statistic	GBE Controller counters	Notes
etherStatsDropEvents	MPC + RNBC	
etherStatsOctets	TOTL + TOTH	
etherStatsPkts	TPR	
etherStatsBroadcastPkts	BPRC	
etherStatsMulticastPkts	MPRC	The Controller doesn't count FC packets
etherStatsCRCAlignErrors	CRCERRS + ALGNERRC	
etherStatsUndersizePkts	RUC	
etherStatsOversizePkts	ROC	
etherStatsFragments	RFC	Should count bad aligned fragments as well
etherStatsJabbers	RJC	Should count bad aligned jabbers as well
etherStatsCollisions	COLC	
etherStatsPkts64Octets	PRC64	RMON counts bad packets as well
etherStatsPkts65to127Octets	PRC127	RMON counts bad packets as well
etherStatsPkts128to255Octets	PRC255	RMON counts bad packets as well
etherStatsPkts256to511Octets	PRC511	RMON counts bad packets as well
etherStatsPkts512to1023Octets	PRC1023	RMON counts bad packets as well
etherStatsPkts1024to1518Octets	PRC1522	RMON counts bad packets as well

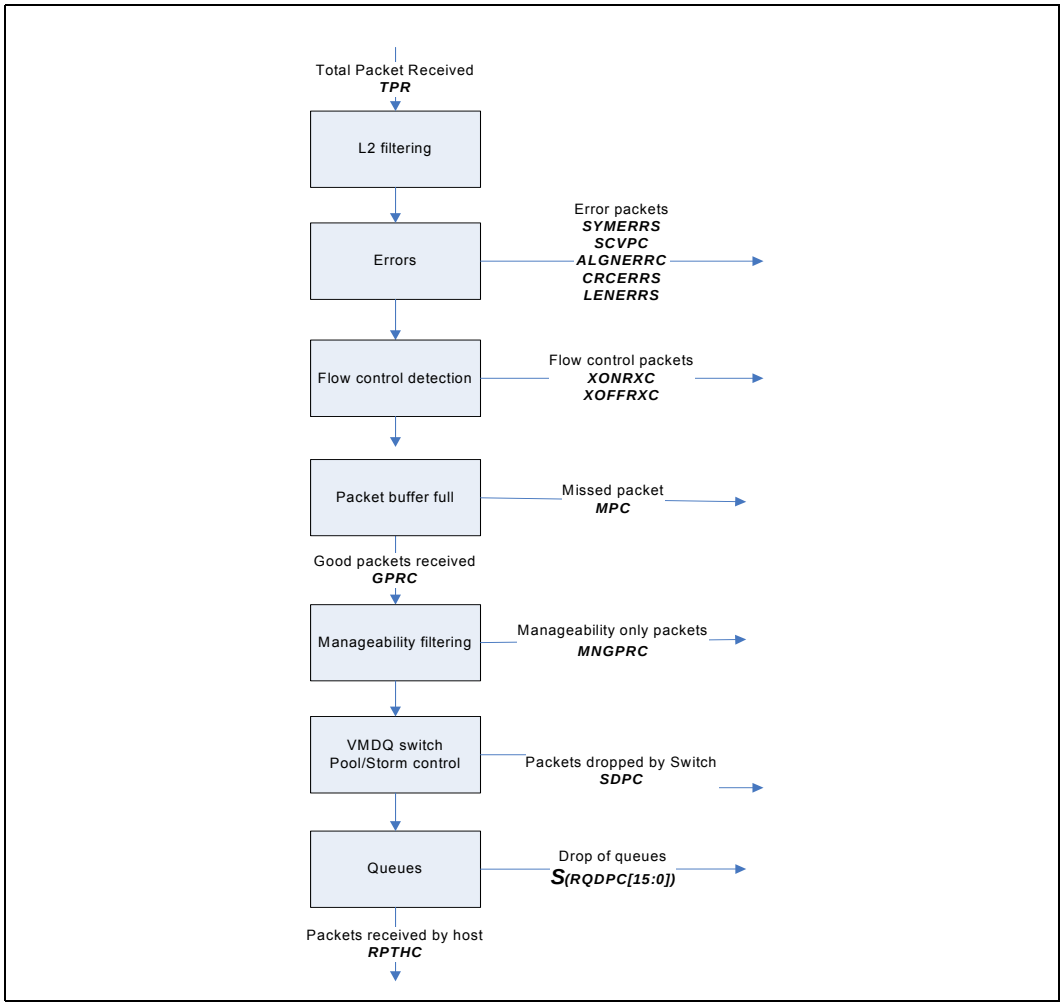
26.10.4 Linux net_device_stats

The Controller supports part of the net_device_stats as defined by Linux Kernel version 2.6 (defined in <linux/netdevice.h>). The following table describes the matching between the internal statistics and the counters requested by this structure.

Table 26-69. Linux net_device_stats

net_device_stats field	GBE Controller counters	Notes
rx_packets	GPRC	The Controller doesn't count flow controls - can be accounted for by using the XONRXC and XOFFRXC counters
tx_packets	GPTC	The Controller doesn't count flow controls - can be accounted for by using the XONTXC and XOFFTXC counters
rx_bytes	GORCL + GORCH	
tx_bytes	GOTCL + GOTCH	
rx_errors	CRCERRS + RLEC + RXERRC + ALGNERRC	
tx_errors	ECOL + LATECOL	
rx_dropped	N/A	
tx_dropped	N/A	
multicast	MPTC	
collisions	COLC	
rx_length_errors	RLEC	
rx_over_errors	N/A	
rx_crc_errors	CRCERRS	
rx_frame_errors	ALGNERRC	
rx_fifo_errors	HRMPC	
rx_missed_errors	MPC	
tx_aborted_errors	ECOL	
tx_carrier_errors	N/A	
tx_fifo_errors	N/A	
tx_heartbeat_errors	N/A	
tx_window_errors	LATECOL	
rx_compressed	N/A	
tx_compressed	N/A	

Figure 26-34.


Total Packet Received
TPR
L2 filtering
Errors
Error packets
SYMERRS
SCVPC
ALGNERRC
CRCERRS
LENERRS
Flow control detection
Flow control packets
XONRXC
XOFFRXC
Packet buffer full
Missed packet
MPC
Good packets received
GPRC
Manageability filtering
Manageability only packets
MNGPRC
VMDQ switch
Pool/Storm control
Packets dropped by Switch
SDPC
Queues
Drop of queues
S(RQDPC[15:0])
Packets received by host
RPTHC

§ §

27.0 GbE Platform Manageability

27.1 Platform Configurations

This section describes the hardware configurations for platform management. It describes the partitioning of platform manageability among system components and the functionality provided by the GbE Controller in each of the platform configurations.

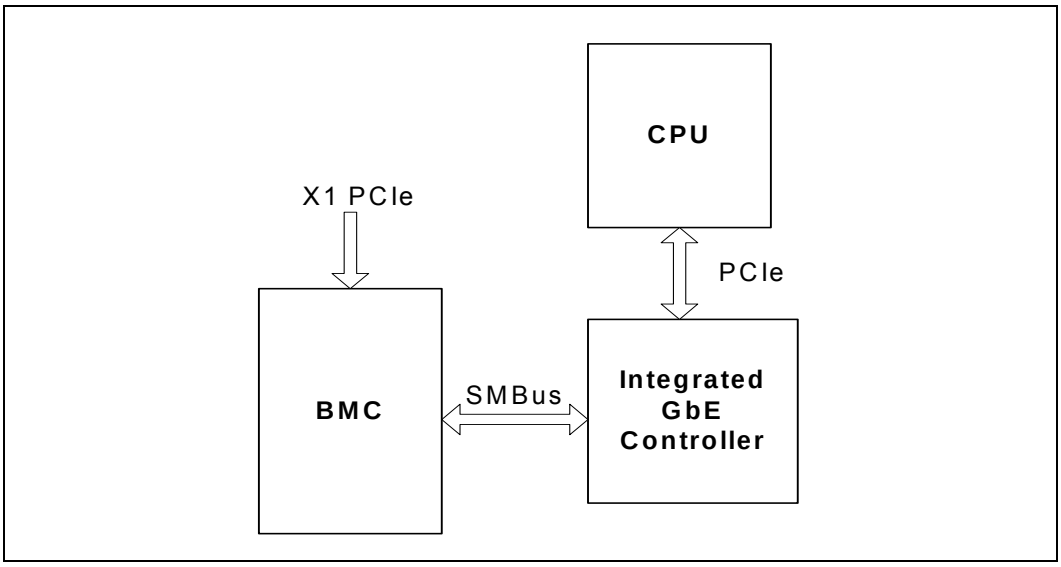
The Controller interfaces with an on-board BMC component for basic manageability functionality. The link between the Controller and the BMC is SMBus.

Note: Link configuration is loaded from the "Redirection LAN Interface" EEPROM bits in the Manageability Capability/Manageability Enable word (word 0x54 - see Section 24.7.5, "Manageability Capability/Manageability Enable (Word 0x54)").

27.1.1 On-Board BMC Configurations

The following figure shows a SMBus-only connection between the Controller and the BMC. All communication between the Controller and the BMC (pass-through traffic, configuration, and status) pass through a single interface. The protocol details for this configuration are according to the SMBus commands described in Section 27.2.1.

Figure 27-1. Controller to BMC Connectivity Through a SMBus-only Connection



See following sections for description of traffic types that use the SMBus interfaces.

27.1.2 GbE Controller

The GbE Controller on a PCIe* I/F may be remotely configured through the SMBus interface.

27.2 Pass Through Functionality

The Controller supports traffic pass through to an external BMC. The usable bandwidth for either direction is up to 400 Kbps in SMBus mode. The following list describes the different flows that can make up pass through traffic:

- BMC management traffic
- Keyboard or mouse traffic for KVM (low data rate)
- Video traffic for KVM (low average rate of 150 Kbytes/s to 200 Kbytes/s) - transmit only
- USB* 2.0 redirect (up to 50 Mb/s)
- IDE redirect for remote CD/floppy (rate - priority 1 - CDx7 = 1.05 Mb/s. Priority 2 - CDx24 = 64 Mb/s
- Serial Over LAN (SoL) - 300 Kbytes/s

Pass through traffic is carried through a SMBus (legacy devices) based on the configuration loaded from the EEPROM. The management interface does not change dynamically and is loaded only during a the Controller power up.

27.2.1 SMBus Pass Through (PT) Functionality

In addition to carrying pass through traffic, in SMBus mode the Controller enables loading default configuration of filters by EEPROM. When working in SMBus mode the default values of the manageability receive filters can be set according to the PT LAN structure defined in Chapter 24.0, “Non-Volatile Memory Map - EEPROM”.

27.2.1.1 Pass Through (PT) Modes

In pass through mode the manageability traffic is handled by the BMC. The Controller is presented on the manageability link as four different devices (via different SMBus addresses) where each device is connected to a different LAN port. There is no logical connection between the devices. Fail-over if required between the LAN ports is done by the external BMC (by sending/receiving packets through different ports). The status reports to the BMC, ARP handling, DHCP and other pass through functionality are unique for each port).

27.3 Programming Interfaces

27.3.1 SMBus Programming

This section describes the SMBus transactions supported in Advanced Pass Through (APT) mode.

27.3.1.1 Write SMBus Transactions (BMC to GbE Controller)

Table 27-1 summarizes the SMBus write transactions supported by the Controller.

Table 27-1. SMBus Write Transactions

TCO Command	Transaction	Command		Fragmentation	Section
Transmit Packet	Block Write	First: Middle: Last:	0x84 0x04 0x44	Multiple	27.3.1.1.1
Transmit Packet	Block Write	Single:	0xC4	Single	27.3.1.1.1
Receive Enable	Block Write	Single:	0xCA	Single	27.3.1.2
Management Control	Block Write	Single:	0xC1	Single	27.3.1.2.2
Update MNG RCV filter parameters	Block Write	Single:	0xCC	Single	27.3.1.2.3
Force TCO	Block Write	Single:	0xCF	Single	27.3.1.2.1
Request Status	Block Write	Single:	0xDD	Single	27.3.1.1.2

27.3.1.1.1 Transmit Packet Command

The Transmit Packet command behavior is detailed in section 22.2.1.1.4. The Transmit Packet fragments have the following format.

Function	Command	Byte Count	Data 1	...	Data N
Transmit first fragment	0x84	N	Packet data MSB	...	Packet data LSB
Transmit middle fragment	0x04				
Transmit last fragment	0x44				
Transmit single fragment	0xC4				

The payload length is limited to the maximum payload length set in the EEPROM.

If the overall packet length is larger than 1536 bytes or smaller than 32 bytes (including padding), then the packet is discarded by the Controller and Abort is asserted.

27.3.1.1.2 Request Status Command

The TCO controller can initiate a request to read the Controller manageability status by sending this command.

When it receives this command, the Controller initiates a notification to the external controller (when it is ready with the status), and then the external controller will be able to read the status, by issuing a read status command (see section 27.3.1.3.3).

Request Status Command format:

Function	Command	Byte Count	Data 1
Request status	0xDD	1	0

27.3.1.2 Receive Enable Command

The Receive Enable command is a single fragment command that is used to configure the Controller.

This command has two formats:

- Short: 1-byte legacy format (providing backward compatibility with previous components).
- Long: 14-byte advanced format (allowing greater configuration capabilities).

Note: If the receive enable command is short and thus does not include all the parameters, then the parameters will be taken from most recent previous configuration (either the most recent long Receive-Enable command in which the particular value was set, or the EEPROM if there was no such previous long Receive-Enable command).

Func.	Cmd	Byte Count	Data 1	Data 2	...	Data 7	Data 8	...	Data 11	Data 12	Data 13	Data 14
Legacy receive enable	0xCA	1	Receive control byte	-	...	-	-	...	-	-	-	-
Advanced receive enable		14 0x0E		MAC addr. MSB		MAC addr. LSB	RSV		RSV	BMC SMBus addr.	Interf. data byte	Alert value byte

Where:

- Receive control byte (data byte 1):

Field	Bit(s)	Description
RCV_EN	0	Receive TCO enable. 0b = Disable Receive and Transmit of TCO packets 1b = Enable Receive and Transmit of TCO packets. Setting this bit will enable BMC to network and network to BMC traffic and all manageability receive filtering operations (Will set bit *MANC.RCV_TCO_EN*). The enable of the specific filtering is done through a special configuration command (see Section 27.3.1.2.3). Notes: 1. When this bit is cleared all receive TCO functionality is disabled, not just the packets that are directed to the BMC. 2. Bit can also be set using the *Enable TCO to Network* bit in the *Update MNG RCV Filter Parameters* command (Parameter Filter Enable).
Reserved	1	Reserved
EN_STA	2	Enable Status reporting to the BMC, on status change. 0b = Disable status reporting 1b = Enable status reporting
Reserved	3	Reserved, must be 0b.

Field	Bit(s)	Description
NM	4-5	Notification Method. Define the notification method the Controller will use. 00b = SMBus Alert 01b = Asynchronous Notify 10b = Direct Receive 11b = Not Supported. **Note:** In multiple SMBus address mode, all SMBus addresses must be configured to the same notification method.
Reserved	6	Reserved, must be 1b.
CBDM	7	Configure BMC dedicated MAC Address. **Note:** This bit should be zero when the RCV_EN bit (bit 0) is not set. 0b = The Controller will share MAC address for MNG traffic with host MAC address which is specified in EEPROM words 0x0-0x2. 1b = The Controller will use the BMC dedicated MAC address as filter for incoming receive packets. The BMC MAC address is set in bytes 2-7 in this command. If short version of the command is used, the Controller will use MAC address configured in the most recent long version of the command in which the CBDM bit was set. If no such previous long command exists, then the Controller will use MAC address 1 (Mac address programmed in MMAL/H1 registers). When Dedicated MAC address feature is activated, the Controller will use the following registers to filter in all the traffic addressed to the BMC MAC. BMC should not modify these registers: Manageability Decision Filter – MDEF7 (and corresponding bit 7 in Management Only traffic Register – MNGONLY) Manageability MAC Address Low – MMAL1 Manageability MAC Address High – MMAH1

- MNG MAC address (data bytes 2-7)
  Ignored if CBDM bit is not set. This MAC address will be used for configuration of the dedicated MAC address. This MAC address will continue to be used when CBDM bit is set in subsequent short versions of this command.
- Reserved (data bytes 8-11).
- Asynchronous Notification SMBus address (data byte 12)
  This address will be used for the Asynchronous Notification SMBus transaction and for Direct Receive.
- Interface data (data byte 13)
  Interface data byte to be used in Asynchronous Notification.
- Alert data (data byte 14).

Alert Value data byte to be used in the Asynchronous Notification.

27.3.1.2.1 Force TCO Command

Depending on the bit set in the TCO mode field this command will cause the Controller to perform either:

1. TCO Reset, if Force TCO reset is enabled in the EEPROM. The Force TCO reset will clear the data-path (RX/TX) of the Controller to enable the BMC to transmit/receive packets through the Controller.
   — The Controller will consider Force TCO reset command as an indication that the OS is hung and will clear the DRV_LOAD flag. If TCO reset is disabled in EEPROM, the Controller clears the *CTRL_EXT.DRV_LOAD* bit but does not reset the data-path.
   — Following TCO reset management sets *MANC.TCO_RESET* to 1.

2. TCO Isolate, If TCO Isolate is enabled in the EEPROM. The TCO Isolate command will disable PCIe* write operations to the LAN port.
   — If TCO Isolate is disabled in the EEPROM, the controller will does not execute the command but sends a response to the BMC with a successful completion.
   — Following TCO Isolate management sets *MANC.TCO_Isolate* to 1.
3. Firmware Reset. This command will cause re-initialization of all the manageability functions and re-load of manageability related EEPROM words (for example, Firmware patch code).
   — On reception of Firmware reset command management sets *MANC.FW_RESET* to 1 and increments the FWSM.Reset_Cnt field.

Note: Force TCO reset will effect only the port related to the SMBus address the command was issued to.

Following firmware reset, BMC will need to re-initialize all ports.

Force TCO Command format:

Function	Command	Byte Count	Data 1
Force TCO	0xCF	1	TCO mode

Where TCO Mode field controls the following operations:

Field	Bit(s)	Description
DO_TCO_RST[1]	0	Do TCO reset. 0b = Do Nothing 1b = Perform TCO Reset Note: Setting this bit generates a one time LAN port reset event.
Reserved	1	Reserved
Firmware Reset	2	Reset Manageability and re-load Manageability related EEPROM words. 0b = Do Nothing 1b = Issue Firmware Reset to manageability. Note: Setting this bit generates a one time Firmware reset event. Following Firmware Reset Management related data from EEPROM is loaded.
Reserved	3-7	Reserved (Set to 0x00).

1. TCO Reset operation enabled in EEPROM.

27.3.1.2.2 Management Control

This command is used to set generic manageability parameters. The parameters list is shown in the table below. The command is C1h, which states that it is a management control command. The first data byte is the parameter number and the data after-words (length and content) are parameter specific as shown in the table.

Note: If in the update configuration, the parameter that the BMC sets is not supported by the Controller, the Controller will not NACK the transaction. After the transaction ends, the Controller will discard the data and will assert a transaction abort status.

This is the format of the Management control command:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Management control	0xC1	N	Parameter number	Parameter dependent		

27.3.1.2.3 Update MNG RCV Filter Parameters

This command is used to set the manageability receive filters parameters. The parameters list is shown in the table below. The command is 0xCC, which states that it is a parameter update. The first data byte is the parameter number and the data after-words (length and content) are parameter specific as shown in the table.

If in the update configuration, the parameter that the BMC sets is not supported by the Controller, it will not NACK the transaction. After the transaction ends, the Controller will discard the data and will assert a transaction abort status.

Detailed description of receive filtering capabilities and configuration is in Section 27.4.

This is the format of the Update MNG RCV filter parameters command:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Update MNG RCV filter parameters	0xCC	N	Parameter number	Parameter dependent		

The table below shows the different parameters and their contents.

Parameter	#	Parameter Data
Filters Enable	0x1	Defines generic filters configuration. The structure of this parameter is 4 bytes. Where Data 2 is the MSB and Data 5 is the LSB. The parameter updates the MANC register. The general filter enable is in the receive enable command, which enables receive filtering. This parameter specifies which filters should be enabled. ARP filtering and dedicated MAC address can also be enabled through the receive enable command (see Section 27.3.1.2).
MNGONLY configuration	0xF	This parameter defines which of the packets types identified as manageability packets in the receive path will never be directed to the host memory. Data 2:5: MNGONLY register bytes - Data 2 is the MSB
Flex filter 0 enable MASK and length	0x10	Flex filter 0 mask. This parameter defines the Mask bits in the FTFT filter. Data 2:17 – MASK. Bit 0 in data 2 is the first bit of the MASK Data 18:19 – Reserved. Should be zero. Data 20 – Flexible Filter length. Must be greater than or equal one.
Flex filter 0 data	0x11	Data 2 – Group of Flex filter's bytes. This parameter defines the Data pattern bytes in the FTFT filter: 0x0 = bytes 0-29 0x1 = bytes 30-59 0x2 = bytes 60-89 0x3 = bytes 90-119 0x4 = bytes 120-127 Data 3:32 – Flex filter data bytes. Data 3 is LSB. Group's length is not mandatory 30 bytes; it may vary according to filter's length and must NOT be padded by zeros.
VLAN Filters	0x62	Three bytes to load the VLAN tag filters (MAVTV) Data 2 – VLAN Filter number Data 3 – MSB of VLAN Filter Data 4 – LSB of VLAN Filter

Parameter	#	Parameter Data
Flex Ports Filters	0x63	Three bytes to load the manageability flex port filters (MFUTP). Data 2 – Flex Port Filter number Data 3 – MSB of flex port filter Data 4 - LSB of flex port filter
IPv4 Filters	0x64	Five bytes to load the IPv4 address filter (MIPAF, DW 15:12) Data 2 – IPv4 address filter number (0-3) Data 3 – LSB of IPv4 address filter … Data 6 – MSB of IPv4 address filter
IPv6 Filters	0x65	17 bytes to load IPv6 address filter (MIPAF) Data 2 – IPv6 address filter number (0-3) Data 3 – LSB of IPv6 address filter … Data 18 – MSB of IPv6 address filter
MAC Filters	0x66	Seven bytes to load MAC address filters (MMAL, MMAH) Data 2 – MAC address filters pair number (0-1) Data 3 – MSB of MAC address … Data 8 – LSB of MAC address
Ethertype Filters	0x67	Five bytes to load Ethertype filters (METF) Data 2 - METF filter index (valid values are 0,) Data 3 - MSB of METF ... Data 6 - LSB of METF
Extended Decision Filter	0x68	Nine bytes to load the extended decision filters (MDEF_EXT & MDEF) Data 2 - MDEF filter index (valid values are 0...6) Data 3 - MSB of MDEF_EXT (DecisionFilter1) .... Data 6 - LSB of MDEF_EXT (DecisionFilter1) Data 7 - MSB of MDEF (DecisionFilter0) .... Data 10 - LSB of MDEF (DecisionFilter0) The command shall overwrite any previously stored value.

27.3.1.3 Read SMBus Transactions (Controller to BMC)

The table below summarizes the different SMBus read transactions supported by the Controller. The read transactions are compatible with the SMBus Read Block Protocol format.

TCO Command	Transaction	Command	Op-Code		Fragmentation	Section
Receive TCO Packet	Block Read	0xC0 or 0xD0	First: Middle: Last[1]	0x90 0x10 0x50	Multiple	27.3.1.3.1
Read configuration	Block Read	0xC6 + 0xD2	Single:	0xD2	Single	27.3.1.3.5
Read Receive Enable configuration	Block Read	0xDA	Single:	0xDA	Single	27.3.1.3.7
Read the Controller Status	Block Read	0xC0 or 0xD0 or 0xDE	Single:	0xDD	Single	27.3.1.3.3

TCO Command	Transaction	Command	Op-Code		Fragmentation	Section
Read Management parameters	Block Read	0xD1	Single:	0xD1	Single	27.3.1.3.5
Read MNG RCV filter parameters	Block Read	0xCD	Single:	0xCD	Single	27.3.1.3.6
Get system MAC address	Block Read	0xD4	Single	0xD4	Single	27.3.1.3.4

1. Last fragment of the receive TCO packet is the packet status.

Notes: 0xC0/0xD0 commands are used for more than one payload. If the BMC issues these read commands and the Controller has no pending data to transfer, it will always return as default opcode 0xDD with the Controller status, and will not NACK the transaction.

27.3.1.3.1 Receive TCO LAN Packet Transaction

The BMC uses this command to read the packet received on the LAN and its status. When the Controller has a packet to deliver to the BMC, it asserts the SMBus notification, for the BMC to read the data. Upon receiving notification of the arrival of LAN receive packet, the BMC should begin issuing a Receive TCO packet command using the block read protocol. The packet can be delivered in more than one SMBus fragment (at least two - one for the packet, and the other one for the status), and the BMC should follow the "F" and "L" bits (2 MSB bits of the op-code).

The op-code can have these values:

- 0x90 - First Fragment
- 0x10 - Middle Fragment.
- 0x50 - Packet status (last fragment) as described below in Section 27.3.1.3.2.

The Receive packet will be silently discarded if following packet reception and notification no valid BMC transaction on the SMBus is detected (no receive fragment is read by the BMC, no Status Read operation is executed by the BMC) within a timeout period. The timeout period is set according to the SMBus notification timeout EEPROM parameter.

Function	Command
Receive TCO packet	0xC0 or 0xD0

Data returned from the Controller:

Function	Byte Count	Data 1 (Op-Code)	Data 2	...	Data N
Receive TCO First Fragment	N	90	Packet Data Byte	...	Packet Data Byte
Receive TCO Middle Fragment		10			
Receive TCO Last Fragment (TCO status, see Section 27.3.1.3.2)		50			

27.3.1.3.2 Receive TCO LAN Status Payload Transaction

This is the last transaction that the Controller issues when a packet that was received from the LAN is transferred to the BMC. The transaction contains the status of the received packet.

This is the format of the status transaction:

Function	Byte Count	Data 1 (Op-Code)	Data 2 – Data 17 (Status data)
Receive TCO long status	17 (0x11)	0x50	See below

The status is 16 bytes where byte 0 (bits 0-7) is set in Data 2 of the status and byte 15 is set in Data 17 of the status.

The table below describes the content of the status data:

Table 27-2. TCO LAN Packet Status Data

Name	Bits	Description
Packet Length	13:0	Packet length including CRC, only 14 LSB bits.
Packet status	35:14	See Table 27-3
Reserved	42:36	Reserved
Error	47:43	See Table 27-4
VLAN	63:48	The two bytes of the VLAN header tag.
Reserved	67:64	Reserved
Packet type	80:68	See Table 27-6
Reserved	84:81	Reserved
MNG status	127:85	See Table 27-7. This field should be ignored if Receive TCO is not enabled,

The meaning of the bits inside of each field is in Section 26.1.5.

Table 27-3. Packet Status Info

Field	Bit(s)	Description
LAN#	21:20	Indicates the source port of the packet
VP	19	VLAN Stripped –insertion of VLAN TAG is needed.
VEXT	18	Additional VLAN present in packet
Reserved	17:15	Reserved
Reserved	13:12	Reserved
CRC stripped	11	Insertion of CRC is needed.
Reserved	10:6	Reserved
UDPV	5	UDP checksum valid
Reserved	4:3	Reserved
IPCS	2	Ipv4 Checksum Calculated on packet
L4I	1	L4 (TCP/UDP) Checksum calculated on packet
UDPCS	0	UDP checksum calculated on packet

Table 27-4. Error Status Info

Field	Bit(s)	Description
RXE	4	RX Data Error
IPE	3	Ipv4 Checksum Error
L4E	2	L4 (TCP/UDP) Checksum Error
Reserved	1:0	Reserved

Table 27-6. Packet Type

Bit Index	Bit 11 = 0b	Bit 11 = 1b (L2 packet)
12	VLAN packet indication	
11	Packet matched one of the ETQF filters.	
9:8	Reserved	Reserved
7	NFS - NFS header present	Reserved
6	SCTP - SCTP header present	
5	UDP - UDP header present	
4	TCP - TCP header present	
3	IPV6E- IPv6 Header includes extensions	
2	IPV6 - IPv6 header present	EtherType - ETQF register index that matches the packet. Special types might be defined for 1588, 802.1X, or any other requested type.
1	IPV4E - IPv4 Header includes extensions	
0	IPV4 - IPv4 header present	

Table 27-7. MNG Status

Name	Bits	Description
Decision Filter match	42:35	Set when there is a match to one of the Decision filters
IPv4/IPv6 match	34	Set when there is an IPv4 match or IPv6 match. This bit is valid only if the bit 30 (IP match bit) or bit 4 (ARP match bit) are set.
IP address match	33	Set when there is a match to any of the IP address filters
IP address Index	32:31	Set when there is a match to the IP filter number. (IPv4 or IPv6)
Flex TCO filter match	30	Set when there is a match to the Flex port filter
Reserved	29:27	Reserved
L4 port match	26	Set when there is a match to any of the UDP / TCP port filters
L4 port Filter Index	25:19	Indicate the flex filter number
Unicast Address match	18	Set when there is a match to any of the 2 Unicast MAC addresses.
Unicast Address Index	17:15	Indicates which of the 2 Unicast MAC addresses match the packet. Valid only if the Unicast Address match is set.
MNG VLAN Address Match	14	Set when the MNG packet matches one of the MNG VLAN filters
Pass MNG VLAN Filter Index	13:11	Indicates which of the Vlan filters match the packet.
Reserved	10:8	Reserved
Pass ARP req / ARP resp	7	Set when the MNG packet is an ARP response/request packet

Table 27-7. MNG Status (Continued)

Name	Bits	Description
Pass MNG neighbor	6	Set when the MNG packet is a neighbor discovery packet.
Pass MNG broadcast	5	Set when the MNG packet is a broadcast packet
Pass RMCP 0x0298	4	Set when the UDP/TCP port of the MNG packet is 0x298
Pass RMCP 0x026F	3	Set when the UDP/TCP port of the MNG packet is 0x26F
Manageability Ethertype filter passed	2	Indicates that one of the METF filters matched
manageability Ethertype filter index	1:0	Indicates which of the METF filters matched

27.3.1.3.3 Read Status Command

The BMC can read the Controller status. The Controller will assert an alert prior to the BMC reading the status bytes. There can be two reasons for the Controller to send status to the BMC:

- The external BMC asserts a request for reading the Controller status (Section 22.2.1.1.2).
- The Controller detects a change in one of the "Status Data 1" bits, and was set to send status to the BMC, on status change, in the receive enable command (Section 22.2.1.1.2).

Commands C0h/D0h are for backward compatibility. 0xD0/0xC0 can be used for other payloads the Controller will define in the op-code, which payload this transaction will be. When 0xDE command is set, the Controller will always return opcode 0xDD with the Controller status. The BMC will read the event causing the notification, using the read status command shown below:

Function	Command
Read status	0xC0 or 0xD0 or 0xDE

Function	Byte Count	Data 1 (Op-Code)	Data 2 (Status Data 1)	Data 3 (Status Data 2)
Receive TCO partial status	3	0xDD	See below	

The table below shows Status Data 1 byte:

Bit	Name	Description
7	LAN Port Lsb	When Lan Port Msb is 0: 0b = Status came from LAN port 0. 1b = Status came from LAN port 1. When Lan Port Msb is 1: 0b = Status came from LAN port 2. 1b = Status came from LAN port 3.
6	TCO command aborted	0b = A TCO command abort event has not occurred since the last Read Status cycle. 1b = A TCO command abort event has occurred since the last Read Status cycle. See Section 22.2.1.1.5 for command abort flow.

5	Link Status indication	0b = LAN link down 1b = LAN link is up
4	Reserved	
3	Initialization indication	0b = An EEPROM reload event has not occurred since the last Read Status cycle. 1b = An EEPROM reload event has occurred since the last Read Status cycle. See note 1.
2	LAN Port Msb	Defines together with LAN Port Lsb the port that sent the Status: Lan Port Msb / Lan Port Lsb 0 / 0 — Status came from LAN port 0. 0 / 1 — Status came from LAN port 1. 1 / 0 — Status came from LAN port 2. 1 / 1 — Status came from LAN port 3.
1:0	Power state	00b = Dr state 01b = D0u state 10b = D0 state 11b = D3 state

Notes:

1. This indication will be asserted when the Controller manageability block will reload EEPROM and its internal data-base will be updated to EEPROM default values. This is an indication that the external BMC should re-configure the Controller, if other values beside the EEPROM default should be configured.

Status data byte 2 is used by the BMC as an indication whether the LAN driver is alive and running.

The driver valid indication is a bit that is set by the driver when it is coming up, and cleared when it goes down to Dx state or cleared by the HW on PCI reset.

Bits 2 and 1 indicate that the LAN driver is not stuck. Bit 2 indicates whether the interrupt line of the LAN function is asserted, and bit 1 indicates whether driver between the last Read Status Cycle dealt the interrupt line.

The table below shows Status Data 2 byte:

Bit	Name	Description
7	Reserved	Reserved
6	Reserved	Reserved
5	Reserved	Reserved
4	Reserved	Reserved
3	Driver Valid indication	0b = LAN driver is not alive 1b = LAN driver is alive
2	Interrupt pending indication	0b = LAN interrupt is not asserted. 1b = LAN interrupt is asserted.
1	ICR register read/write	0b = ICR register was not read since the last Read Status Cycle. 1b = ICR register was read since the last Read Status cycle. Reading the ICR means that the driver has dealt with the interrupt that was asserted
0	Reserved	Reserved

The table below shows the possible values of bits 2, 1 and what the BMC can assume based on their value:

Previous	Current	
Don't care	00b	Interrupt is not pending – O.K
00b	01b	New interrupt is asserted – O.K
10b	01b	New interrupt is asserted – O.K
11b	01b	Interrupt is waiting for reading – O.K
01b	01b	Interrupt is waiting for reading by driver more than one read status Cycle – Not OK (possible driver hang state).
Don't Care	11b	Previous interrupt was read and current interrupt is pending – O.K
Don't Care	10b	Interrupt is not pending – O.K

Note: The BMC reads should consider the time it takes for the driver to deal with the interrupt (few micro-seconds), too frequent reads will give false indications.

27.3.1.3.4 Get System MAC Address

The Get System MAC Address will return system MAC Address over the SMBus. This command is a single fragment Read Block transaction, which will return the following database:

The Get System MAC Address returns the system MAC Address (RAL0, RAH0).

Get System MAC Address format:

Function	Command
Get system MAC address	0xD4

Data returned from the Controller:

Function	Byte Count	Data 1 (Op-Code)	Data 2	...	Data 7
Get system MAC address	7	0xD4	MAC address MSB	…	MAC address LSB

27.3.1.3.5 Read Management Parameters

To read the management parameters the BMC should execute two SMBus transactions: a block write that sets the parameter that the BMC wants to read, and then a block read that reads the parameter.

This is the block write transaction:

Function	Command	Byte Count	Data 1
Management control Request	0xC1	1	Parameter number

Following the block write the BMC should issue a block read that will read the parameter that was set in the block write command:

Function	Command
Read management parameter	0xD1

Data returned from the Controller:

Function	Byte Count	Data 1 (Op-Code)	Data 2	Data 3	...	Data N
Read management parameter	N	0xD1	Parameter number	Parameter dependent		

The returned data is as follows:

Parameter	#	Parameter Data
Reserved	0x00	A single byte parameter: Data 3 - Bit [7:0] Reserved
Wrong parameter request	0xFE	Returned by the Controller only. This parameter is returned on read transaction, if in the previous read command the BMC sets a parameter that is not supported by the Controller.
Controller ot ready	0xFF	Returned by the Controller only, on read parameters command when the data that should have been read is not ready. This parameter has no data. The BMC should retry the read transaction.

Notes: The parameter that is returned might not be the parameter requested by the BMC. The BMC should verify the parameter number (default parameter to be returned is 1h).

It is BMC's responsibility to follow the procedure defined above. If the BMC sends a block read command (as described above) that is not preceded by a block write command with bytecount=1, the Controller will set the Parameter Number in the read block transaction to 0xFE.

27.3.1.3.6 Read MNG RCV Filter Parameters

To read the MNG RCV filter parameters, the BMC should execute two SMBus transactions: a block write that sets the parameter that the BMC wants to read, and then a block read that reads the parameter.

This is the block write transaction:

Function	Command	Byte Count	Data 1	Data 2
MNG RCV filter parameters to read	0xCC	1 or 2	Parameter number	Parameter data

The table below shows the different parameters and their contents:

Parameter	#	Parameter Data
Filters Enable	0x1	None
MNGONLY configuration	0xF	None
Flex filter 0 enable MASK and length	0x10	None
Flex filter 0 data	0x11	Data 2 – Group of Flex filter's bytes: 0x0 = bytes 0-29 0x1 = bytes 30-59 0x2 = bytes 60-89 0x3 = bytes 90-119 0x4 = bytes 120-127
VLAN Filters	0x62	One byte to define the accessed VLAN tag filter (MAVTV) Data 2 – VLAN Filter number
Flex Ports Filters	0x63	One byte to define the accessed manageability flex port filter (MFUTP). Data 2 – Flex Port Filter number
IPv4 Filter	0x64	One byte to define the accessed IPv4 address filter (MIPAF) Data 2 – IPv4 address filter number
IPv6 Filters	0x65	One byte to define the accessed IPv6 address filter (MIPAF) Data 2 – IPv6 address filter number
MAC Filters	0x66	One byte to define the accessed MAC address filters pair (MMAL, MMAH) Data 2 – MAC address filters pair number (0-1)
Ethertype Filters	0x67	One byte to define Ethertype filters (METF) Data 2 - METF filter index (valid values are 0,1,2 and 3.)
Extended Decision Filter	0x68	One byte to define the extended decision filters (MDEF_EXT & MDEF) Data 2 - MDEF filter index (valid values are 0...6)
Wrong parameter request	0xFE	Returned by the Controller only. This parameter is returned on read transaction, if in the previous read command the BMC sets a parameter that is not supported by the Controller.
Controller is not ready	0xFF	Returned by the Controller only, on read parameters command when the data that should have been read is not ready. This parameter has no data.

Following the block write the BMC should issue a block read that will read the parameter that was defined in the block write command:

Function	Command
Request MNG RCV filter parameters	0xCD

Data returned from the Controller:

Function	Byte Count	Data 1 (Op-Code)	Data 2	Data 3	...	Data N
Read MNG RCV filter parameters	N	0xCD	Parameter number	Parameter dependent		

The returned data is in the same format of the "update" command 1.

Note: If the returned parameter is not the parameter that the BMC requested, then the BMC should verify the parameter number (default parameter to be returned is 1h).

- If the parameter number is 0xFF, it means that the data that the Controller should supply is not yet ready. The BMC should retry the read transaction.
- It is BMC's responsibility to follow the above procedure. If the BMC sends a block read command (as described above) that is not preceded by a block write command, then the Controller will set Parameter Number in the read block transaction to be 0xFE.

27.3.1.3.7 Read Receive Enable Configuration

The BMC uses this command to read the receive configuration data. This data can be configured in receive enable command or through EPROM loading upon power-up.

Read Receive Enable Configuration command format (SMBus Read Block Protocol):

Function	Command
Read receive enable	0xDA

Data returned from the Controller.

Function	Byte Count	Data 1 (Op-Code)	Data 2	Data 3	...	Data 8	Data 9	...	Data 12	Data 13	Data 14	Data 15
Read receive enable	15 (0x0F)	0xDA	Receive control byte	MAC address MSB	...	MAC address LSB	IP address MSB	...	IP address LSB	BMC SMBus address	Interface data byte	Alert value byte

The detailed description of each field is specified in the receive enable command description in Section 27.3.1.2.

27.3.1.4 SMBus ARP Transactions

All SMBus-ARP transactions include PEC byte.

27.3.1.4.1 Prepare to ARP

This command will clear the Address Resolved flag (set to false). It will not affect the status or validity of the dynamic SMBus Address (will not clear the address Valid flag). It is used to inform all devices that the ARP Master is starting the ARP process:

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	0000 0001	0	[Data dependent value]	0	

27.3.1.4.2 Reset Device (General)

This command will clear the Address Resolved flag (set to false). It will not affect the status or validity of the dynamic SMBus Address (will not clear the address Valid flag).

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	0000 0010	0	[Data dependent value]	0	

27.3.1.4.3 Reset Device (Directed)

The Command field is NACK-ed if the bits 7 through 1 do not match the current the Controller SMBus address. It clears the Address Resolved flag (set to false). It will not affect the status or validity of the dynamic SMBus Address (will not clear the address Valid flag).

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	Targeted slave address | 0	0	[Data dependent value]	0	

27.3.1.4.4 Assign Address

This command assigns the Controller SMBus address. The address and command bytes are always acknowledged.

The transaction is aborted immediately (NACK-ed) if any of the UDID bytes differ from the Controller UDID bytes as defined in Section 22.2.1.1.7. If successful, the MNG will update the SMBus address internally. This command will also set the Address Resolved flag to true.

1	7	1	1	8	1	8	1	
S	Slave Address	Wr	A	Command	A	Byte Count	A	•••
	1100 001	0	0	0000 0100	0	0001 0001	0	

8	1	8	1	8	1	8	1	
Data-1	A	Data-2	A	Data-3	A	Data-4	A	•••
UDID byte 15 (MSB)	0	UDID byte 14	0	UDID byte 13	0	UDID byte 12	0	

8	1	8	1	8	1	8	1	
Data-5	A	Data-6	A	Data-7	A	Data-8	A	•••
UDID byte 11	0	UDID byte 10	0	UDID byte 9	0	UDID byte 8	0	

8	1	8	1	8	1	
Data-9	A	Data-10	A	Data-11	A	•••
UDID byte 7	0	UDID byte 6	0	UDID byte 5	0	

8	1	8	1	8	1	8	1	
Data-12	A	Data-13	A	Data-14	A	Data-15	A	•••
UDID byte 4	0	UDID byte 3	0	UDID byte 2	0	UDID byte 1	0	

8	1	8	1	8	1	1
Data-16	A	Data-17	A	PEC	A	P
UDID byte 0 (LSB)	0	Assigned Address	0	[Data dependent value]	0	

27.3.1.4.5 Get UDID (General and Directed)

The Get UDID command depends on whether this is a directed or general command. The General Get UDID SMBus transaction supports a constant command value of 0x03.

The Directed Get UDID SMBus transaction supports a dynamic command value equal to the dynamic SMBus address with the LSB bit set.

Note: Bit 0 (LSB) of Data byte 17 will always be 1b.

If the SMBus Address has been resolved (Address Resolved flag is true), then for a general command the MNG will not acknowledge (NACK) this transaction, or for a directed command the MNG will always acknowledge (ACK) this transaction.

This command does not affect the status or validity of the dynamic SMBus Address (will not clear the address Valid flag) nor of the Address Resolved flag.

The command returns the UDID bytes as defined in Section 22.2.1.1.7.

S	Slave Address	Wr	A	Command	A	S	•••
	1100 001	0	0	See below	0		

7	1	1	8	1	
Slave Address	Rd	A	Byte Count	A	•••
1100 001	1	0	0001 0001	0	

8	1	8	1	8	1	8	1	
Data-1	A	Data-2	A	Data-3	A	Data-4	A	•••
UDID byte 15 (MSB)	0	UDID byte 14	0	UDID byte 13	0	UDID byte 12	0	

8	1	8	1	8	1	8	1	
Data-5	A	Data-6	A	Data-7	A	Data-8	A	•••
UDID byte 11	0	UDID byte 10	0	UDID byte 9	0	UDID byte 8	0	

8	1	8	1	8	1	
Data-9	A	Data-10	A	Data-11	A	•••
UDID byte 7	0	UDID byte 6	0	UDID byte 5	0	

8	1	8	1	8	1	8	1	
Data-12	A	Data-13	A	Data-14	A	Data-15	A	•••
UDID byte 4	0	UDID byte 3	0	UDID byte 2	0	UDID byte 1	0	

8	1	8	1	8	1	1
Data-16	A	Data-17	A	PEC	~Ã	P
UDID byte 0 (LSB)	0	Device Slave Address	0	[Data dependent value]	1	

27.4 Manageability Receive Filtering

27.4.1 Overview and General Structure

This section describes the manageability receive packet filtering flow. The description applies to each of the Controller LAN ports. A packet that is received by the Controller can have one of the following results:

- Discarded
- Sent to host memory
- Sent to the external BMC
- Sent to both the BMC and host memory

The decisions regarding forwarding of packets to the host and to the BMC are separate and are configured through two sets of registers. However, the BMC may define some types of traffic as exclusive. This traffic will be forwarded only to the BMC, even if it passes the filtering process of the host. These types of traffic are defined using the MNGONLY register.

The BMC controls the types of packets that it receives by programming the receive manageability filters. The software device driver can not write to the manageability filter registers. The table below lists the registers that are only written by the BMC.

Table 27-8. Registers Written by the BMC (Sheet 1 of 2)

Register	Functionality	When Reset
MANC	General configuration of the manageability filters.	LAN_PWR_GOOD and Firmware Reset
MNGONLY	Enables routing of packets exclusively to the manageability.	LAN_PWR_GOOD and Firmware Reset
MDEF[7:0] MDEF_EXT[7:0]	Configuration of manageability decision filters	LAN_PWR_GOOD and Firmware Reset
MMAH[1:0], MMAL[1:0]	Two Unicast MAC manageability addresses	LAN_PWR_GOOD
MAVTV[7:0]	Eight VLAN tag values	LAN_PWR_GOOD
MFUTP[3:0]	Eight UDP/TCP destination port values	LAN_PWR_GOOD

Table 27-8. Registers Written by the BMC (Sheet 2 of 2)

Register	Functionality	When Reset
FTFT	Values and mask and length for the 1 flex TCO filter	LAN_PWR_GOOD
MIPAF	IP address for manageability filtering	LAN_PWR_GOOD
METF	L2 EtherType values	LAN_PWR_GOOD

These registers are reset only on LAN_PWR_GOOD. In SMBus PT mode, registers that enable filters or functionality are loaded from the EEPROM following a firmware reset. See Section 24.10 for description of their location in the EEPROM map.

The high-level structure of manageability filtering is done using two steps.

1. The packet is parsed and fields in the header are compared to programmed filters.
2. A set of decision filters are applied to the result of the first step.

General rules:

- Fragmented packets are passed to manageability but not parsed beyond the IP header.
- Packets with L2 errors (CRC, alignment, etc.) are never forwarded to manageability.

The following sections describe the manageability filtering, followed by the final filtering rules. The filtering rules are created by programming the decision filters as described in Section 27.4.4.

27.4.2 L2 Filters

27.4.2.1 MAC and VLAN Filters

The manageability MAC filters allow comparison of the Destination MAC address to one of two filters defined in the *MMAH* and *MMAL* registers.

The VLAN filters allow comparison of the 12-bit VLAN tag to one of eight filters defined in the *MAVTV* registers.

27.4.2.2 EtherType Filters

The manageability L2 EtherType filters allow filtering of receive packets based on the Layer 2 EtherType field. The L2 type field of incoming packets is compared against the EtherType filters programmed in the *METF.EType* (up to 4 filters) and the result is incorporated to the decision filters.

Each of the manageability EtherType filters can be configured as pass (“positive”) or reject (“negative”) polarity. When there is no match on its EtherType, and the polarity bit is set for a filter, that filter is enabled so as to participate in the decision process.

Note: For the reverse polarity mode to be effective and block certain type of packets, the ethertype filter should be part of all the enabled decision filters.

27.4.3 L3 and L4 Filters

27.4.3.1 ARP Filtering

The Controller supports filtering of both ARP request packets (initiated externally) and ARP responses (to requests initiated by the BMC or the Controller).

27.4.3.2 Neighbor Discovery Filtering

The Controller supports filtering of neighbor Discovery packets. neighbor Discovery uses the IPV6 destination address filters defined in the MIPAF registers (all enabled IPv6 addresses are matched for neighbor Discovery).

27.4.3.3 RMCP Port Filtering

The Controller supports reception of RMCP packets by enabling filtering of the fixed destination ports numbers, port 0x26F and port 0x298.

27.4.3.4 Flex Port Filtering

The Controller implements 8 flex UDP/TCP destination port filters. The Controller directs packets whose L4 destination port matches the value of the respective word in the *MFUTP* registers. The BMC must ensure that only valid entries are enabled in the decision filters below.

27.4.3.5 The Controller IP Address Filtering

The Controller supports filtering by IP address through IPv4 and IPv6 address filters, dedicated to manageability. Two modes are possible, depending on the value of the MANC. EN_IPv4_FILTER bit:

- EN_IPv4_FILTER = 0b: The Controller provides four IPv6 address filters.
- EN_IPv4_FILTER = 1b: The Controller provides three IPv6 address filters and four IPv4 address filters.

27.4.3.6 Checksum Filter

If bit MANC.EN_XSUM_FILTER is set, the Controller directs packets to the BMC only if they match all other filters previously described as well as pass L3/L4 checksum (if it exists).

27.4.4 Manageability Decision Filters

The manageability decision filters are a set of eight filters with the same structure. The filtering rule for each decision filter is programmed by the BMC. The filtering rule defines which of the manageability filters participate in the decision. A packet that passes at least one rule is directed to manageability.

The manageability filters are controlled by the BMC only and not by the LAN driver. To filter network traffic according to a filtering rule, the BMC should define the filtering rule by programming the *MDEF* and *MDEF_EXT* registers and set the *MDEF_EXT.apply_to_network_traffic* bit to 1, to enable the decision filter.

The inputs to each decision filter are:

- Packet passed one of the management L2 unicast address filter.
- Packet is a broadcast packet.

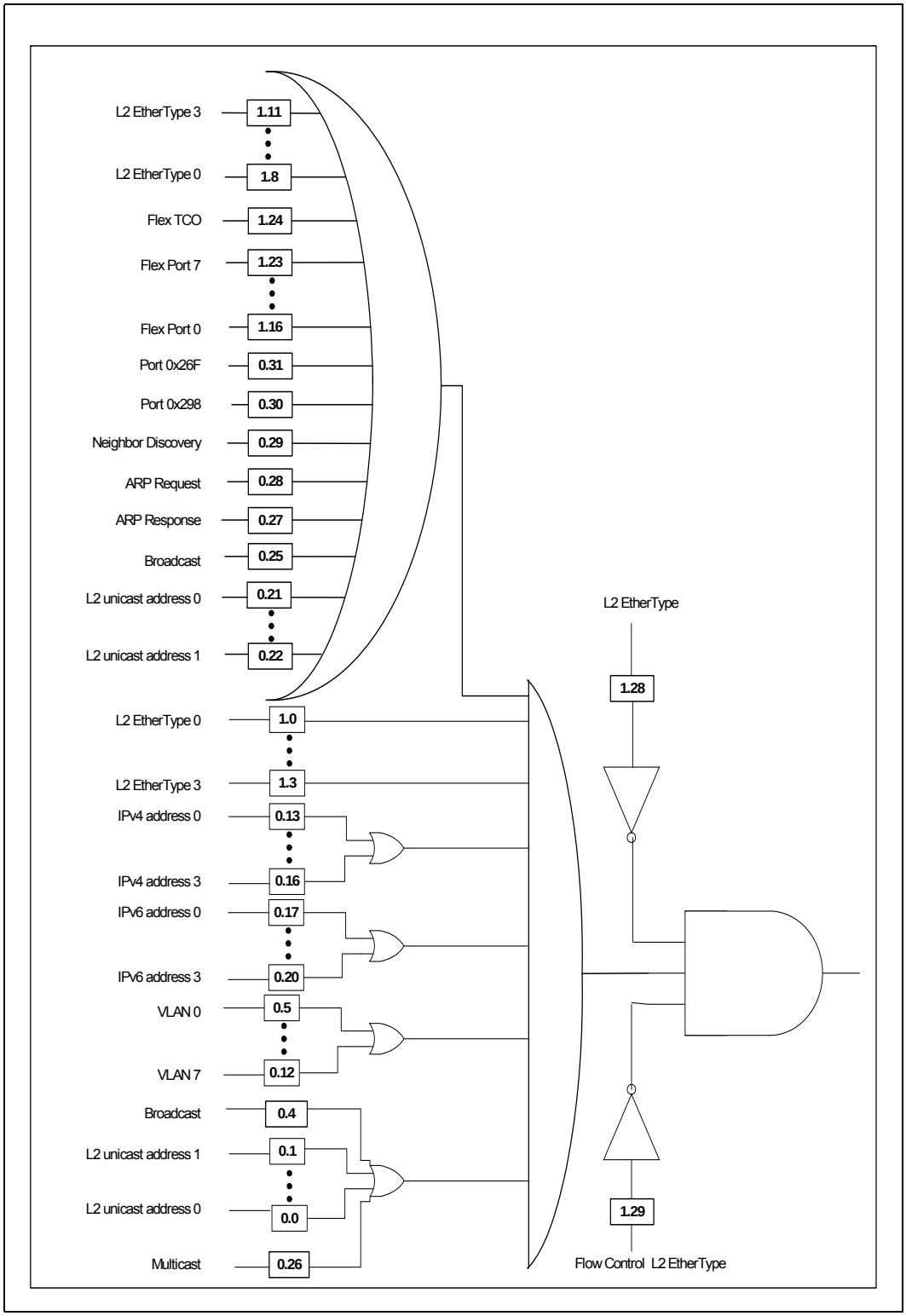
- Packet has a VLAN header and it passed one of the manageability VLAN filters.
- Packet matched one of the IPv4 or IPv6 manageability address filters.
- Packet is a multicast packet.
- Packet passed ARP filtering (request or response).
- Packet passed neighbor Discovery filtering.
- Packet passed 0x298/0x26F port filter.
- Packet passed a valid flex port filter.
- Packet passed a valid flex TCO filter.
- Packet passed or failed an L2 EtherType filter.

The structure of each of the decision filters is shown in Section 27-2, "Manageability Decision Filters". A boxed "x.y" number indicates that the input is conditioned on a mask bit "y" defined in the *MDEF* and *MDEF_EXT* registers for this rule (where x=0 denotes MDEF and x=1 denotes MDEF_EXT). The decision filter rules are as follows:

- All enabled AND filters must match for the decision filter to match. An AND filter not enabled in the MDEF/MDEF_EXT registers is ignored. In a set of AND filters from the same type, it is enough that one of the enabled filter match. The AND filter types are:
  - L2 address - includes the Unicast AND, Broadcast AND and Multicast AND filters.
  - VLAN - includes all the VLAN AND filters.
  - IPv4 - includes all the IPv4 AND filters.
  - IPv6 - includes all the IPv6 AND filters.
- If at least one OR filter is enabled in the respective *MDEF* and *MDEF_EXT* registers, then at least one of the enabled OR filters must match for the decision filter to match. Otherwise, the OR filters are ignored in the decision (the filter might still match).

A decision filter (for any of the eight filters) defines which of the above inputs is enabled as part of the rule. The BMC programs two 32-bit registers per rule (*MDEF[7:0]* and *MDEF_EXT[7:0]*) with the settings as described in Chapter 28.0, "GbE Programming Interface". A set bit allows its corresponding filter to participate in the filtering decision.

Figure 27-2. Manageability Decision Filters



27.4.4.1 Exclusive Traffic

The decisions regarding forwarding of packets to the host for LAN traffic or to the LAN for host traffic are independent from the management decision filters. However, the BMC may define some types of traffic as exclusive. The behavior for such traffic is defined by the using the bits corresponding to the decision filter in the *MNGONLY* register (one bit per each of the eight decision rules) and the *MDEF_EXT.apply_to_network_traffic* bit. Table 27-9 describes the behavior in each case. If one or more filters match the traffic and at least one of the filters is set as exclusive, the traffic is treated as exclusive.

Table 27-9. Exclusive Traffic Behavior

Filter match		Filter doesn't match
MNGONLY = 0	**MNGONLY = 1**	**N/A**
Traffic is forwarded to the manageability. Traffic is forwarded to the host according to host filtering	Traffic is forwarded only to manageability.	Traffic is forwarded to the host according to host filtering

27.4.5 Possible Configurations

This section describes ways of using management filters. Actual usage may vary.

27.4.5.1 Dedicated MAC Packet Filtering

- Select one of the eight rules for dedicated MAC filtering.
- Load Host MAC address to one of the management MAC address filters and set the appropriate bit in field 1:0 of the MDEF register.
- Set other bits to qualify which packets are allowed to pass through. For example:
  - Load one or more management VLAN filters and set the appropriate bits in field 12:5 of the MDEF register to qualify the relevant manageability VLANs.
  - Set relevant bits in field 20:13 of the MDEF register to qualify with a match to one of the IP addresses.
  - Set any L3/L4 bits (bits 31:27 in the MDEF register and bits 23:16 in the MDEF_EXT register) to filter using any set of L3/L4 filters.

27.4.5.2 Broadcast Packet Filtering

- Select one of the eight rules for broadcast filtering.
- Set bit 25 in the MDEF register of the decision rule to enforce broadcast filtering.
- Set other bits to qualify which broadcast packets are allowed to pass through. For example:
  - Set bit 5 in the MDEF register to filter with the first manageability VLAN.
  - Set relevant bits in field 20:13 of the MDEF register to qualify with a match to one of the IP addresses.
  - Set any L3/L4 bits (bits 31:27 in the MDEF register and bits 23:16 in the MDEF_EXT register) to filter with any set of L3/L4 filters.

27.4.5.3 VLAN Packet Filtering

- Select one of the eight rules for VLAN filtering. Set bit 2 of the decision rule to enforce VLAN filtering.
- Load one or more management VLAN filters and set the appropriate bits in field 12:5 of the MDEF register to qualify the relevant manageability VLANs.
- Set other bits to qualify which VLAN packets are allowed to pass through. For example: Set any L3/L4 bits (bits 31:27 in the MDEF register and bits 23:16 in the MDEF_EXT register) to filter using appropriate L3/L4 filter set.

27.4.5.4 IPv6 Filtering

IPv6 filtering is done using the following IPv6-specific filters:

- IP Unicast filtering — requires filtering for Link Local address and a Global address. Filtering setup might depend on whether or not the MAC address is shared with the Host or dedicated to manageability:
  - — Dedicated MAC address (for example, dynamic address allocation with DHCP does not support multiple IP addresses for one MAC address). In this case, filtering can be done at L2 using two dedicated unicast MAC filters.
  - — Shared MAC address (for example, static address allocation sharing addresses with Host). In this case, filtering needs to be done at L3, requiring two IPv6 address filters, one per address.
- A neighbor Discovery filter — Supports IPv6 neighbor Discovery protocol. Since the protocol relies on multicast packets, Supports filtering of these packets. IPv6 multicast addresses are translated into corresponding Ethernet multicast addresses in the form of 33-33-xxxx-xx-xx, where the last 32 bits of address are taken from the last 32 bits of the IPv6 multicast address. As a result, two direct MAC filters can be used to filter IPv6 solicited-node multicast packets and IPv6 all node multicast packets.

27.4.5.5 Receive Filtering with Shared IP - CPMP

When using the SMBus interface, it is possible to share the Host MAC and IP address with the BMC.

When the BMC shares the MAC and IP address with the Host, receive filtering is based on identifying specific flows through port allocation. The following setting might be used:

- Select one of the eight rules for Dedicated MAC filtering.
- Load Host MAC address to one of the management MAC address filters and set the appropriate bit in field 1:0 of the MDEF register to enforce MAC address filtering using the MAC address.
- If VLAN is used for management, load one or more management VLAN filters and set the appropriate bits in field 12:5 of the MDEF register to qualify the relevant manageability VLANs.
- ARP filter/Neighbor Discovery filter is enabled when the BMC is responsible for handling the ARP protocol. Set bit 27 or bit 28 in the MDEF register for this functionality.
- Set other bits to qualify which packets are allowed to pass through. For example: Set any L3/L4 bits (bits 31:27 in the MDEF register and bits 23:16 in the MDEF_EXT register) to filter using the appropriate L3/L4 filters.

§ §

28.0 GbE Programming Interface

28.1 Introduction

This chapter details the programmer visible state inside the GbE Controller. In some cases, it describes hardware structures invisible to software to clarify a concept. The controllers's address space is mapped into four regions with PCI Base Address Registers. These regions are listed in Table 28-1.

Table 28-1. Address Space Regions

Addressable Content	How Mapped	Size of Region
Internal registers and memories ("Memory BAR")	Direct memory-mapped	128K
Internal registers and memories	I/O Window mapped	32 bytes (3)
MSI-X (optional)	Direct memory-mapped	16K

The internal registers and memories can be accessed though I/O space indirectly. The internal register/memory space is described in the following sections. The PHY registers are accessed through the MDIO interface.

28.1.1 Memory, I/O Address, and Configuration Decoding

28.1.1.1 Memory-Mapped Access to Internal Registers and Memories

The internal registers and memories might be accessed as direct memory-mapped offsets from the base address register (BAR0 or BAR 0/1). See Section 28.1.3 for the appropriate offset for each specific internal register.

28.1.1.2 Memory-Mapped Access to MSI-X Tables

The MSI-X tables may be accessed as direct memory-mapped offsets from the base address register (BAR3). See Section 28.1.3 for the appropriate offset for each specific internal MSIX register.

28.1.1.3 I/O-Mapped Access to Internal Registers and Memories

To support pre-boot operation (prior to the allocation of physical memory base addresses), all internal registers and memories can be accessed using I/O operations. I/O accesses are supported only if an I/O Base Address is allocated and mapped (BAR2), the BAR contains a valid (non-zero value), and I/O address decoding is enabled in the PCIe* configuration.

When an I/O BAR is mapped, the I/O address range allocated opens a 32-byte "window" in the system I/O address map. Within this window, two I/O addressable registers are implemented: IOADDR and IODATA. The IOADDR register is used to specify a reference to an internal register or memory, and then the IODATA register is used as a "window" to the register or memory address specified by IOADDR:

Table 28-2. IOADDR and IODATA in I/O Address Space

Offset	Abbreviation	Name	RW	Size
0x00	IOADDR	Internal Register or Internal Memory location address. 0x00000-0x1FFFF – Internal Registers and Memories 0x20000-0xFFFFFFFF – Undefined	RW	4 bytes
0x04	IODATA	Data field for reads or writes to the Internal Register or Internal Memory Location as identified by the current value in IOADDR. All 32 bits of this register are read/write-able.	RW	4 bytes
0x08 – 0x1F	Reserved	Reserved	RO	4 bytes

28.1.1.3.1 IOADDR (I/O Offset 0x00)

The IOADDR register must be written as a DWORD access. Writes of less than 32 bits are ignored. Reads of any size return a DWORD of data; however, the chipset or CPU might only return a subset of that DWORD.

For software programmers, the IN and OUT instructions must be used to cause I/O cycles to be used on the PCIe* bus. Because writes must be to a 32-bit quantity, the source register of the OUT instruction must be EAX (the only 32-bit register supported by the OUT command). For reads, the IN instruction can have any size target register, but it is recommended that the 32-bit EAX register be used.

Because only a particular range is addressable, the upper register bits are hard coded to zero. Bits 30 through 20 are not write-able and always read back as 0b. Set Bit 31 (*IOADDR.Configuration IO Access Enable*) of the *IOADDR* register to 1.

At hardware reset (LAN_PWR_GOOD) or PCI Reset, this register value resets to 0x00000000. Once written, the value is retained until the next write or reset.

28.1.1.3.2 IODATA (I/O Offset 0x04)

The IODATA register must be written as a DWORD access when the IOADDR register contains a value for the Internal Register and Memories (for example, 0x00000-0x1FFFC). In this case, writes that are less than 32 bits are ignored.

Reads to IODATA of any size returns a DWORD of data. However, the chipset or CPU might only return a subset of that DWORD.

For software programmers, the IN and OUT instructions must be used to cause I/O cycles to be used on the PCIe* bus. Where 32-bit quantities are required on writes, the source register of the OUT instruction must be EAX (the only 32-bit register supported by the OUT command).

Writes and reads to IODATA when the IOADDR register value is in an undefined range (0x20000-0xFFFFFFFC) should not be performed. Results cannot be determined.

Notes: There are no special software timing requirements on accesses to IOADDR or IODATA. All accesses are immediate, except when data is not readily available or acceptable. In this case, the Controller delays the results through normal bus methods (for example, split transaction or transaction retry).

Because a register/memory read or write takes two IO cycles to complete, software must provide a guarantee that the two IO cycles occur as an atomic operation. Otherwise, results can be non-deterministic from the software viewpoint.

28.1.1.3.3 Undefined I/O Offsets

I/O offsets 0x08 through 0x1F are considered to be reserved offsets with the I/O window. Dword reads from these addresses returns 0xFFFF; writes to these addresses are discarded.

28.1.1.4 Configuration Access to Internal Registers and Memories

To support 'legacy' pre-boot 16-bit operating environments without requiring IO address space, the Controller enables accessing CSRs via configuration address space by mapping the *IOADDR* and *IODATA* registers into configuration address space. If the *CSR_conf_en* bit in the *PCIe* Init Configuration 2* EEPROM word (Word 0x19) is set to 1, access to CSRs via Configuration address space is enabled. The registers mapping in this case is shown in Table 28-3.

Table 28-3. IOADDR and IODATA in Configuration Address Space

Configuration Address	Abbreviation	Name	RW	Size
0x98	IOADDR	Internal register or internal memory location address. 0x00000-0x1FFFF – Internal Registers and Memories 0x20000-0x7FFFFF – Undefined	RW	4 bytes
0x9C	IODATA	Data field for reads or writes to the internal register or internal memory location as identified by the current value in IOADDR. All 32 bits of this register are read/write-able.	RW	4 bytes

Software writes data to an internal CSR via configuration space in the following manner:

1. CSR address is written to the IOADDR register where:
   a. Bit 31 (*IOADDR.Configuration IO Access Enable*) of the *IOADDR* register should be set to 1.
   b. Bits 30:0 of *IOADDR* should hold the actual address of the internal register or memory being written to.
2. Data to be written is written into the IODATA register.
   — The *IODATA* register is used as a "window" to the register or memory address specified by *IOADDR* register. As a result the data written to the *IODATA* register is written into the CSR pointed to by bits 30:0 of the *IOADDR* register.
3. *IOADDR.Configuration IO Access Enable* is cleared, to avoid un-intentional CSR read operations (that may cause clear by read) by other applications scanning the configuration space.

Software reads data from an internal CSR via Configuration space in the following manner:

1. CSR address is written to the IOADDR register where:
   a. Bit 31 (*IOADDR.Configuration IO Access Enable*) of the The *IOADDR* register should be set to 1.
   b. Bits 30:0 of *IOADDR* should hold the actual address of the internal register or memory being read.
2. CSR value is read from the *IODATA* register.
   a. The *IODATA* register is used as a "window" to the register or memory address specified by *IOADDR* register. As a result the data read from the *IODATA* register is the data of the CSR pointed to by bits 30:0 of the *IOADDR* register

3. *IOADDR.Configuration IO Access Enable* is cleared, to avoid un-intentional CSR read operations (that may cause clear by read) by other applications scanning the configuration space.

Notes:

- If the *CSR_conf_en* bit in the *PCIe* Init Configuration 2* EEPROM word is cleared, then accesses to the *IOADDR* and *IODATA* registers via the configuration address space are ignored and have no effect on the register and the CSRs referenced by the *IOADDR* register.
- When Function is in D3 state Software should not attempt to access CSRs via the *IOADDR* and *IODATA* Configuration registers.
- To enable CSR access via configuration space, Software should set to 1 bit 31 (*IOADDR.Configuration IO Access Enable*) of the IOADDR register. Software should clear bit 31 of the IOADDR register after completing CSR access to avoid an unintentional "clear by read" operation, by another application scanning the configuration address space.
- Bit 31 of the *IOADDR* register (*IOADDR.Configuration IO Access Enable*) has no effect when initiating access via IO Address space.

28.1.2 Register Conventions

All registers in the Controller are defined to be 32 bits and should be accessed as 32-bit double-words, with the exceptions of register pairs where two 32 bit registers make up a larger logical size.

Some registers contain bits that are marked as "reserved". Software should never set these bits to a value of "1." Reads from registers containing reserved bits might return indeterminate values in the reserved bit-positions unless read values are explicitly stated. When read, software should ignore these reserved bits.

Any register address not explicitly declared in this specification should be considered as reserved, and should not be written to. Writing to reserved or undefined register addresses might cause indeterminate behavior. Reads from reserved or undefined configuration register addresses might return indeterminate values unless read values are explicitly stated for specific addresses.

Most registers define the initial hardware values prior to being programmed. In some cases, hardware initial values are undefined and is listed as such via the text "undefined", "unknown", or "X". Such configuration values might need to be set via EEPROM configuration or via software in order for proper operation to occur; this need is dependent on the function of the bit. Other registers might cite a hardware default which is overridden by a higher-precedence operation. Operations which might supersede hardware defaults might include a valid EEPROM load, completion of a hardware operation (such as hardware auto-negotiation), or writing of a different register whose value is then reflected in another bit.

For registers that should be accessed as 32-bit double words, partial writes (less than a 32-bit double word) does not take effect. The write is ignored. Partial reads returns all 32 bits of data regardless of the byte enables.

Note: Partial reads to read-on-clear registers (*ICR*) can have unexpected results since all 32 bits are read regardless of the byte enables. Partial reads should not be done.

- All statistics registers are implemented as 32-bit registers. Though some logical statistics registers represent counters in excess of 32 bits in width, registers must be accessed using 32-bit operations (for example, independent access to each 32-bit field). When reading 64 bits statistics registers the least significant 32-bit register should be read first.

See the special notes for VLAN Filter Table, Multicast Table Arrays and Packet Buffer Memory that are in the specific register definitions.

The Controller register fields are assigned one of the attributes described in Table 28-4.

Table 28-4. GbE Controller Register Field Attributes

Attribute	Description
RW	Read-Write field: Register bits are read-write and can be either set or cleared by software to the desired state.
RWS	Read-Write Status field: Register bits are read-write and can be either set or cleared by software to the desired state. However, the value of this field might be changed by the hardware to reflect a status change.
RO	Read-only register: Register bits are read-only and should not be altered by software. Register bits might be initialized by hardware mechanisms such as serial EEPROM or reflect a status of the hardware state.
R/W1C	Read-only status, Write-1-to-clear status register: Register bits indicate status when read, a set bit indicating a status event can be cleared by writing a 1b. Writing a 0b to R/W1C bit has no effect.
Rsv	Reserved. Write 0 to these fields and ignore read.
RC	Read-only status, Read-to-clear status register: Register bits indicate status when read, a set bit indicating a status event is cleared by reading it.
SC	Self Clear field: a command field that is self clearing. These field are always read as zero.
WO	Write only field: a command field that can not be read, These field read values are undefined.
RC/W	Read-Write status, Read-to-clear status register: Read-to-clear status register. Register bits indicate status when read. Register bits are read-write and can be either set or cleared by software to the desired state.
RC/W1C	Read-only status, Write-1-to-clear status register: Read-to-clear status register. Register bits indicate status when read, a set bit indicating a status event can be cleared by writing a 1b or by reading the register. Writing a 0b to RC/W1C bit has no effect.
RS	Read Set – This is the attribute used for Semaphore bits. These bits are set by read in case the previous values were zero. In this case the read value is zero; otherwise the read value is one. Cleared by write zero.

Note: For all binary equations appearing in the register map, the symbol "|" is equivalent to a binary OR operation.

28.1.2.1 Registers Byte Ordering

This section defines the structure of registers that contain fields carried over the network. Some examples are L2, L3 and L4 fields.

The following example is used to describe byte ordering over the wire (hex notation):

```
Last                    First
...,06, 05, 04, 03, 02, 01, 00
```

Each byte is sent with the LSbit first. That is, the bit order over the wire for this example is

```
Last                                        First
..., 0000 0011, 0000 0010, 0000 0001, 0000 0000
```

The general rule for register ordering is to use Host Ordering (also called little endian). Using the above example, a 6-byte fields (MAC address) is stored in a CSR in the following manner:

```
                  Byte 3 Byte 2 Byte 1 Byte0
DW address (N)    0x03   0x02   0x01   0x00
DW address (N+4)  ...    ...    0x05   0x04
```

The exceptions listed below use network ordering (also called big endian). Using the above example, a 16-bit field (EtherType) is stored in a CSR in the following manner:

```
                  Byte 3 Byte 2 Byte 1 Byte0
(DW aligned)      ...    ...    0x00   0x01
or
(Word aligned)    0x00   0x01   ...    ...
```

The following exceptions use network ordering:

All ETherType fields. For example, the VET EXT field in the VET register, the EType field in the ETQF register, the EType field in the METF register.

Note: The "normal" notation as it appears in text books, etc. is to use network ordering. Example: Suppose a MAC address of 00-A0-C9-00-00-00. The order on the network is 00, then A0, then C9, etc. However, the host ordering presentation would be:

```
                  Byte 3 Byte 2 Byte 1 Byte0
DW address (N)    00     C9     A0     00
DW address (N+4)  ...    ...    00     00
```

28.1.3 Detailed Register Summary

28.1.3.1 PCI Views

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 1 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
0000h	0003h	"Device Control Register—CTRL [0:3] (0x00000; R/W)" on page 1266	0x08100201
00008h	0000Bh	"Device Status Register— STATUS[0:3] (0x0008; R)" on page 1269	0x80080400
00018h	0001Bh	"Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W)" on page 1271	0x00500000

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 2 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
00020h	00023h	"MDI Control Register—MDIC[0:3] (0x0020; R/W)" on page 1273	0x10000000
00E04h	00E07h	"MDC/MDIO Configuration Register—MDICNFG [0:3] (0x0E04; R/W)" on page 1274	0x0
0034h	0037h	"Copper/Fiber Switch Control—CONNSW[0:3] (0x0034; R/W)" on page 1275	0x0
0038h	003Bh	"VLAN Ether Type—VET [0:3] (0x0038; R/W)" on page 1275	0x81008100
E00h	E03h	"LED Control—LEDCTL [0:3](0x0E00; RW)" on page 1276	0x07568302
2404h	2407h	"Internal Receive Packet Buffer Size—IRPBS [0:3] (0x2404; RO)" on page 1277	0x0
3404h	3407h	"Internal Transmit Packet Buffer Size—ITPBS [0:3] (0x3404; RO)" on page 1278	0x8
0010h	0013h	"EEPROM Control Register—EEC [0:3] (0x0010; R/W)" on page 1279	Xh
0014h	0017h	"EEPROM Read Register—EERD [0:3] (0x0014; RW)" on page 1281	0x0
1038h	103Bh	"EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO)" on page 1281	0x0
1060h	1063h	"VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO)" on page 1282	0x0
1010h	1013h	"Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO)" on page 1283	0x80000000
1014h	1017h	"Management EEPROM Read/Write Data—EEMNGDATA [0:3] (0x1014; RO)" on page 1284	0x0
0028h	002Bh	"Flow Control Address Low—FCAL [0:3] (0x0028; RO)" on page 1285	0x00C28001
002Ch	002Fh	"Flow Control Address High—FCAH [0:3] (0x002C; RO)" on page 1286	0x00000100
0030h	0033h	"Flow Control Type—FCT [0:3] (0x0030; R/W)" on page 1286	0x00008808
0170h	0173h	"Flow Control Transmit Timer Value—FCTTV [0:3] (0x0170; R/W)" on page 1287	0x0
2160h	2163h	"Flow Control Receive Threshold Low—FCRTL0 [0:3] (0x2160; R/W)" on page 1288	0x0
2168h	216Bh	"Flow Control Receive Threshold High—FCRTH0 [0:3] (0x2168; R/W)" on page 1289	0x0
2460h	2463h	"Flow Control Refresh Threshold Value—FCRTV[0:3](0x2460; R/W)" on page 1290	0x0
2464h	2467h	"Flow Control Status—FCSTS0 [0:3] (0x2464; RO)" on page 1291	0x0
5B30h	5B33h	"Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO)" on page 1291	0x0
5B64h	5B67h	"Mirrored Revision ID—MREVID[0:3] (0x5B64; R/W)" on page 1293	0x0
5B6Ch	5B6Fh	"PCIe* Control Extended Register—GCR_EXT[0:3] (0x5B6C; R/W)" on page 1294	0x0
5B50h	5B53h	"Software Semaphore—SWSM[0:3] (0x5B50; R/W)" on page 1295	0x0
5B54h	5B57h	"Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS)" on page 1296	0x0
5B5Ch	5B5Fh	"Firmware Synchronization—SW_FW_SYNC[0:3] (0x5B5C; RWS)" on page 1298	0x0
5B04h	5B07h	"Software Mailbox Write—SWMBWR[0:3] (0x5B04; R/W)" on page 1299	0x0
5B08h	5B0Bh	"Software Mailbox 0—SWMB0[0:3] (0x5B08; RO)" on page 1299	0x0
5B0Ch	5B0Fh	"Software Mailbox 1—SWMB1[0:3] (0x5B0C; RO)" on page 1299	0x0
5B18h	5B1Bh	"Software Mailbox 2—SWMB2[0:3] (0x5B18; RO)" on page 1300	0x0
5B1Ch	5B1Fh	"Software Mailbox 3—SWMB3[0:3] (0x5B1C; RO)" on page 1300	0x0
1580h	1583h	"EICR Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)" on page 1301	0x0
1580h	1583h	"EICR Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)" on page 1302	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 3 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
1520h	1523h	"EICS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)" on page 1302	0x0
1520h	1523h	"EICS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)" on page 1303	0x0
1524h	1527h	"EIMS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)" on page 1303	0x80000000
1524h	1527h	"EIMS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)" on page 1304	Xh
1528h	152Bh	"EIMC Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)" on page 1305	0x80000000
1528h	152Bh	"EIMC Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)" on page 1305	0x0
152Ch	152Fh	"Extended Interrupt Auto Clear—EIAC [0:3] (0x152C; R/W)" on page 1306	0x0
1530h	1533h	"EIAM Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)" on page 1307	0x0
1530h	1533h	"EIAM Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)" on page 1307	0x0
1500h	1503h	"Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C)" on page 1308	0x0
1504h	1507h	"Interrupt Cause Set Register—ICS [0:3] (0x1504; WO)" on page 1310	0x0
1508h	150Bh	"Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W)" on page 1312	0x0
150Ch	150Fh	"Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO)" on page 1314	0x0
1510h	1513h	"Interrupt Acknowledge Auto Mask Register—IAM [0:3] (0x1510; R/W)" on page 1316	0x0
1680h at 4	1683h at 4	"Interrupt Throttle—EITR [0:3][0:9] (0x1680 + 4*n [n = 0...9]; R/W)" on page 1317	0x0
1700h at 4	1703h at 4	"Interrupt Vector Allocation Registers—IVAR [0:3][0:3](0x1700 + 4*n [n=0...3]; RW)" on page 1318	0x0
1740h	1743h	"Interrupt Vector Allocation Registers—MISC IVAR_MISC[0:3] (0x1740; RW)" on page 1319	0x0
1514h	1517h	"General Purpose Interrupt Enable—GPIE [0:3] (0x1514; RW)" on page 1320	0x0
5B68h	5B6Bh	"MSIX PBA Clear—PBACL [0:3] (0x5B68; R/W1C)" on page 1324	0X0
0100h	0103h	"Receive Control Register—RCTL[0:3] (0x0100; R/W)" on page 1325	0x0
C00Ch at 0x4	C00Fh at 0x4	"Split and Replication Receive Control—SRRCTL [0:3][0:7] (0xC00C + 0x40*n [n=0...7]; R/W)" on page 1328	0x00000400
C04Ch at 0x40	C04Fh at 0x40	"Split and Replication Receive Control—SRRCTL [0:3][1:7](0xC00C + 0x40*n [n=0...7]; R/W)" on page 1329	0x80000400
5480h at 0x4	5483h at 0x4	"Packet Split Receive Type—PSRTYPE [0:3][0:7] (0x5480 + 4*n [n=0...7]; R/W)" on page 1331	0x0007FFFE
54C0h	54C3h	"Replicated Packet Split Receive Type—RPLPSRTYPE [0:3] (0x54C0; R/W)" on page 1332	0x0007FFFE
C000h at 0x40	C000h at 0x40	"Receive Descriptor Base Address Low—RDBAL [0:3][0:7] (0xC000 + 0x40*n [n=0...7]; R/W)" on page 1333	0x0
C004h at 0x40	C007h at 0x40	"Receive Descriptor Base Address High—RDBAH [0:3][0:7] (0xC004 + 0x40*n [n=0...7]; R/W)" on page 1333	0x0
C008h at 0x40	C00Bh at 0x40	"Receive Descriptor Ring Length—RDLEN [0:3][0:7](0xC008 + 0x40*n [n=0...7]; R/W)" on page 1334	0x0
C010h at 0x40	C013h at 0x40	"Receive Descriptor Head—RDH [0:3][0:7] (0xC010 + 0x40*n [n=0...7]; RO)" on page 1334	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 4 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
C018h at 0x40	C01Bh at 0x40	"Receive Descriptor Tail—RDT [0:3][0:7] (0xC018 + 0x40*n [n=0...7]; R/W)" on page 1335	0x0
C028h at 0x40	C02Bh at 0x40	"Receive Descriptor Control—RXDCTL [0:3][0:7] (0xC028 + 0x40*n [n=0...7]; R/W)" on page 1335	0x0000140C
C030h at 0x40	C033h at 0x40	"Receive Queue Drop Packet Count—RQDPC [0:3][0:7] (0xC030 + 0x40*n [n=0...7]; RC/W)" on page 1337	0x0
5000h	5003h	"Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W)" on page 1338	0x00000300
5004h	5007h	"Receive Long Packet Maximum Length—RLPML [0:3] (0x5004; R/W)" on page 1340	0x00002600
5008h	500Bh	"Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W)" on page 1340	0x1
5200h at 0x4	5203h at 0x4	"Multicast Table Array—MTA [0:3][0:127] (0x5200 + 4*n [n=0...127]; R/W)" on page 1341	0x0
5400h at 0x8	5403h at 0x8	"Receive Address Low 0—RAL0 [0:3][0:15] (0x5400 + 8*n [n=0...15]; 0x54E0 + 8*n [n=0...7]; R/W)" on page 1342	0x0
54E0h at 0x8	54E3h at 0x8	"Receive Address Low 1—RAL1 [0:3][0:7] (0x54E0 + 8*n [n=0...7]; R/W)" on page 1343	0x0
5404h at 0x8	5407h at 0x8	"Receive Address High 0—RAH0 [0:3][0:15] (0x5404 + 8*n [n=0...15]; 0x54E04 + 8*n [n=0...7]; R/W)" on page 1344	0x0
54E4h at 0x8	54E7h at 0x8	"Receive Address High 1—RAH1 [0:3][0:7] (0x54E4 + 8*n [n=0...7]; R/W)" on page 1345	0x0
5600h at 0x4	5603h at 0x4	"VLAN Filter Table Array—VFTA [0:3][0:127] (0x5600 + 4*n [n=0...127]; R/W)" on page 1346	0x0
5818h	581Bh	"Multiple Receive Queues Command Register—MRQC [0:3](0x5818; R/W)" on page 1347	0x0
5C80h at 0x4	5C83h at 0x4	"RSS Random Key Register—RSSRK [0:3][0:9] (0x5C80 + 4*n [n=0...9]; R/W)" on page 1349	0x0
5C00h at 0x4	5C03h at 0x4	"Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W)" on page 1349	0x0
5A80h at 0x4	5A83h at 0x4	"Immediate Interrupt RX—IMIR [0:3][0:7] (0x5A80 + 4*n [n=0...7]; R/W)" on page 1351	0x0
5AA0h at 0x4	5AA3h at 0x4	"Immediate Interrupt Rx Ext.—IMIREXT [0:3][0:7] (0x5AA0 + 4*n [n=0...7]; R/W)" on page 1352	0x0
59E0h at 0x4	59E3h at 0x4	"2tuples Queue Filter—TTQF[0:3][0:7] (0x59E0 + 4*n[n=0..7]; R/W)" on page 1353	0x0
5AC0h	5AC3h	"Immediate Interrupt Rx VLAN Priority—IMIRVP [0:3] (0x5AC0; R/W)" on page 1354	0x0
55FCh	55FFh	"SYN Packet Queue Filter—SYNQF [0:3] (0x55FC; RW)" on page 1354	0x0
5CB0h at 0x4	5CB3h at 0x4	"EType Queue Filter—ETQF [0:3][0:7] (0x5CB0 + 4*n[n=0...7]; RW)" on page 1355	0x0
0400h	0403h	"Transmit Control Register—TCTL [0:3] (0x0400; R/W)" on page 1356	0x000400F8
0404h	0407h	"Transmit Control Extended—TCTL_EXT [0:3] (0x0404; R/W)" on page 1357	0x00010840
0410h	0413h	"Transmit IPG Register—TIPG [0:3] (0x0410; R/W)" on page 1358	0x00601008
041Ch	041Fh	"Retry Buffer Control—RETX_CTL [0:3] (0x041C; R/W)" on page 1359	0x00000003
3590h	3593h	"DMA Tx Control—DTXCTL [0:3] (0x3590; R/W)" on page 1359	0x00400004
359Ch	359Fh	"DMA TX TCP Flags Control Low—DTXTCPFLGL [0:3] (0x359C; R/W)" on page 1360	0x00400004
35A0h	35A3h	"DMA TX TCP Flags Control High—DTXTCPFLGH [0:3] (0x35A0; RW)" on page 1361	0x00000F7F

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 5 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
3540h	3543h	"DMA TX Max Total Allow Size Requests—DTXMXSZRQ [0:3] (0x3540; RW)" on page 1361	0x00000010
355Ch	355Fh	"DMA TX Maximum Packet Size—DTXMXPKTSZ [0:3] (0x355C; RW)" on page 1362	0x00000098
E000h	E003h	"Transmit Descriptor Base Address Low—TDBAL [0:3][0:7] (0xE000 + 0x40*n [n=0...7]; R/W)" on page 1362	0x0
E004h at 0x40	E007h at 0x40	"Transmit Descriptor Base Address High—TDBAH [0:3][0:7] (0xE004 + 0x40*n [n=0...7]; R/W)" on page 1363	0x0
E008h at 0x40	E00Bh at 0x40	"Transmit Descriptor Ring Length—TDLEN [0:3][0:7] (0xE008 + 0x40*n [n=0...7]; R/W)" on page 1363	0x0
E010h at 0x40	E013h at 0x40	"Transmit Descriptor Head—TDH [0:3] [0:7] (0xE010 + 0x40*n [n=0...7]; RO)" on page 1364	0x0
E018h at 0x40	E01Bh at 0x40	"Transmit Descriptor Tail—TDT [0:3][0:7] (0xE018 + 0x40*n [n=0...7]; R/W)" on page 1364	0x0
E028h at 0x40	E02Bh at 0x40	"Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W)" on page 1365	0x0
E038h at 0x40	E03Bh at 0x40	"Tx Descriptor Completion Write-Back Address Low—TDWBAL [0:3][0:7] (0xE038 + 0x40*n [n=0...7]; R/W)" on page 1367	0x0
E03Ch at 0x40	E03Fh at 0x40	"Tx Descriptor Completion Write-Back Address High - TDWBAH [0:3][0:7] (0xE03C + 0x40*n [n=0...7];R/W)" on page 1367	0x0
C014h at 0x40	C017h at 0x40	"Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W)" on page 1368	0x0000A200
E014h at 0x40	E017h at 0x40	"Tx DCA Control Registers—TXCTL [0:3][0:7] (0xE014 + 0x40*n [n=0...7]; R/W)" on page 1370	0x00002A00
5B70h	5B73h	"DCA Requester ID Information—DCA_ID [0:3] (0x5B70; RO)" on page 1371	0x0
5B74h	5B77h	"DCA Control—DCA_CTRL [0:3] (0x5B74; R/W)" on page 1371	0x1
581Ch	581Fh	"VMDq Control Register—VT_CTL [0:3] (0x581C; R/W)" on page 1372	0x0
3558h	355Bh	"Malicious Driver Free Block—MDFB [0:3] (0x3558; RWS)" on page 1373	0x0
3548h	354Bh	"Last VM Misbehavior Cause—LVMMC[0:3] (0x3548; RC)" on page 1373	0x0
5AD0h at 0x4	5AD3h at 0x4	"VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW)" on page 1374	0x80002600
5AF0h	5AF3h	"Replication Offload Register—RPLOLR [0:3] (0x5AF0; RW" on page 1375	0x80000000
5DD0h at 0x4	5DD3h at 0x4	"VLAN VM Filter—VLVF [0:3][0:31] (0x5D00 + 4*n [n=0...31]; RW)" on page 1376	0x0
A000h at 0x4	A003h at 0x4	"Unicast Table Array—UTA [0:3][0:127] (0xA000 + 4*n [n=0...127]; R/W)" on page 1377	0x0
5DB0h	5DB3h	"Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW)" on page 1377	0x00000800
5DB4h	5DB7h	"Storm Control Status—SCSTS [0:3] (0x5DB4;RO)" on page 1378	0x0
5DB8h	5DBBh	"Broadcast Storm Control Threshold—BSCTRH [0:3] (0x5DB8;RW)" on page 1379	0x0
5DBCh	5DBFh	"Multicast Storm Control Threshold—MSCTRH [0:3] (0x5DBC; RW)" on page 1379	0x0
5DC0h	5DC3h	"Broadcast Storm Control Current Count - BSCCNT [0:3] (0x5DC0;RO)" on page 1379	0x0
5DC4h	5DC7h	"Multicast Storm Control Current Count—MSCCNT [0:3] (0x5DC4;RO)" on page 1380	0x0
5DC8h	5DCBh	"Storm Control Time Counter—SCTC [0:3] (0x5DC8; RO)" on page 1380	0x0
5DCCh	5DCFh	"Storm Control Basic Interval—SCBI [0:3] (0x5DCC; RW)" on page 1381	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 6 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
5D80h at 0x4	5D83h at 0x4	"Virtual Mirror Rule Control—VMRCTL [0:3][0:3] (0x5D80 + 0x4*n [n= 0...3]; RW)" on page 1381	0x0
5D90h at 0x4	5D93h at 0x4	"Virtual Mirror Rule VLAN—VMRVLAN [0:3][0:3] (0x5D90 + 0x4*n [n= 0...3]; RW)" on page 1382	0x0
5DA0h at 0x4	5DA3h at 0x4	"Virtual Mirror Rule VM—VMRVM [0:3][0:3] (0x5DA0 + 0x4*n [n= 0...3]; RW)" on page 1382	0x0
2500h	2503h	"DMA Receive Power Saving Register—DMARPS [0:3] (0x2500; R/W)" on page 1383	0x0000010A
2504h	2507h	"DMA Transmit Power Saving Register—DMATPS [0:3] (0x2504; R/W)" on page 1384	0x0000010A
1040h	1043h	"Watchdog Setup—WDSTP [0:3] (0x1040; R/W)" on page 1385	0x01000000
1044h	1047h	"Watchdog Software Device Status—WDSWSTS [0:3] (0x1044; R/W)" on page 1386	0x0
1048h	104Bh	"Free Running Timer—FRTIMER [0:3] (0x1048; RWS)" on page 1386	0x0
104Ch	104Fh	"TCP Timer—TCPTIMER [0:3] (0x104C; R/W)" on page 1387	0x0
B620h	B623h	"RX Time Sync Control Register—TSYNCRXCTL [0:3] (0xB620;RW)" on page 1388	0x0
B624h	B627h	"RX timestamp Low—RXSTMPL [0:3] (0xB624; RO)" on page 1389	0x0
B628h	B62Bh	"RX Timestamp High—RXSTMPH [0:3] (0xB628; RO)" on page 1389	0x0
B62Ch	B62Fh	"RX Timestamp Attributes Low—RXSATRL[0:3] (0xB62C; RO)" on page 1389	0x0
B630h	B633h	"RX timestamp Attributes High—RXSATRH [0:3] (0xB630; RO)" on page 1390	0x0
B614h	B617h	"TX Time Sync Control Register—TSYNCTXCTL [0:3] (0xB614; RW)" on page 1390	0x0
B618h	B61Bh	"TX Timestamp Value Low—TXSTMPL [0:3] (0xB618;RO)" on page 1391	0x0
B61Ch	B61Fh	"TX Timestamp Value High—TXSTMPH[0:3] (0xB61C; RO)" on page 1391	0x0
B6F8h	B6FBh	"System Time Register Residue—SYSTIMR [0:3] (0xB6F8; RW)" on page 1391	0x0
B600h	B603h	"System Time Register Low—SYSTIML [0:3] (0xB600; RW)" on page 1392	0x0
B604h	B607h	"System Time Register High—SYSTIMH [0:3] (0xB604; RW)" on page 1392	0x0
B60Ch	B60Fh	"Time Adjustment Offset Register Low—TIMADJL [0:3] (0xB60C; RW)" on page 1393	0x0
B610h	B613h	"Time Adjustment Offset Register High—TIMADJH [0:3] (0xB610;RW)" on page 1393	0x0
B640h	B643h	"TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW)" on page 1394	0x80000000
B644h	B647h	"Target Time Register 0 Low—TRGTTIML0 [0:3] (0xB644; RW)" on page 1396	0x0
B648h	B64Bh	"Target Time Register 0 High—TRGTTIMH0 [0:3] (0xB648; RW)" on page 1396	0x0
B64Ch	B64Fh	"Target Time Register 1 Low—TRGTTIML1 [0:3] (0xB64C; RW)" on page 1396	0x0
B650h	B653h	"Target Time Register 1 High—TRGTTIMH1 [0:3] (0xB650; RW)" on page 1397	0x0
B654h	B657h	"Frequency Out 0 Control Register—FREQOUT0 [0:3] (0xB654; RW)" on page 1397	0x0
B658h	B65Bh	"Frequency Out 1 Control Register—FREQOUT1 [0:3] (0xB658; RW)" on page 1398	0x0
B65Ch	B65Fh	"Auxiliary Time Stamp 0 Register Low—AUXSTMPL0 [0:3] (0xB65C; RO)" on page 1398	0x0
B660h	B663h	"Auxiliary Time Stamp 0 Register High—AUXSTMPH0 [0:3] (0xB660; RO)" on page 1399	0x0
B664h	B667h	"Auxiliary Time Stamp 1 Register Low—AUXSTMPL1 [0:3] (0xB664; RO)" on page 1399	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 7 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
B668h	B66Bh	"Auxiliary Time Stamp 1 Register High—AUXSTMPH1 [0:3] (0xB668; RO)" on page 1400	0x0
5F50h	5F53h	"Time Sync RX Configuration—TSYNCRXCFG [0:3] (0x5F50; R/W)" on page 1400	0x0
003Ch	003Fh	"Time Sync SDP Configuration Register—TSSDP [0:3] (0x003C; R/W)" on page 1401	0x0
B66Ch	B66Fh	"Time Sync Interrupt Cause Register—TSICR [0:3] (0xB66C; RC/W1C)" on page 1402	0x0
B674h	B677h	"Time Sync Interrupt Mask Register—TSIM [0:3] (0xB674; RW)" on page 1403	0x0
B670h	B673h	"Time Sync Interrupt Set Register—TSIS [0:3] (0xB670; WO)" on page 1404	0x0
4200h	4203h	"PCS Configuration—PCS_CFG [0:3] (0x4200; R/W)" on page 1405	0x00000008
4208h	420Bh	"PCS Link Control—PCS_LCTL [0:3] (0x4208; RW)" on page 1405	0x0204000C
420Ch	420Fh	"PCS Link Status—PCS_LSTS [0:3] (0x420C; RO)" on page 1407	0x0000000C
4218h	421Bh	"AN Advertisement—PCS_ANADV [0:3] (0x4218; R/W)" on page 1409	0x00000020
421Ch	421Fh	"Link Partner Ability—PCS_LPAB [0:3] (0x421C; RO)" on page 1410	0x0
4220h	4223h	"Next Page Transmit—PCS_NPTX [0:3] (0x4220; RW)" on page 1411	0x0
4224h	4227h	"Link Partner Ability Next Page—PCS_LPABNP [0:3] (0x4224; RO)" on page 1412	0x0
01028h	0102Bh	"SFP I2C Command—I2CCMD [0:3] (0x1028; R/W)" on page 1413	0x000000XX
0102Ch	0102Fh	"SFP I2C Parameters—I2CPARAMS [0:3] (0x102C; R/W)" on page 1414	0x00000046
04000h	04003h	"CRC Error Count—CRCERRS [0:3] (0x4000; RC)" on page 1415	0x0
04004h	04007h	"Alignment Error Count—ALGNERRC [0:3] (0x4004; RC)" on page 1416	0x0
04008h	0400Bh	"Symbol Error Count—SYMERRS [0:3] (0x4008; RC)" on page 1416	0x0
04010h	04013h	"Missed Packets Count—MPC [0:3] (0x4010; RC)" on page 1417	0x0
04014h	04017h	"Single Collision Count—SCC [0:3] (0x4014; RC)" on page 1417	0x0
04018h	0401Bh	"Excessive Collisions Count—ECOL [0:3] (0x4018; RC)" on page 1418	0x0
0401Ch	0401Fh	"Multiple Collision Count—MCC [0:3] (0x401C; RC)" on page 1418	0x0
04020h	04023h	"Late Collisions Count—LATECOL [0:3] (0x4020; RC)" on page 1419	0x0
04028h	0402Bh	"Collision Count—COLC [0:3] (0x4028; RC)" on page 1419	0x0
04030h	04033h	"Defer Count—DC [0:3] (0x4030; RC)" on page 1420	0x0
0403Ch	0403Fh	"Host Transmit Discarded Packets by MAC Count—HTDPMC [0:3] (0x403C; RC)" on page 1420	0x0
04040h	04043h	"Receive Length Error Count—RLEC [0:3] (0x4040; RC)" on page 1421	0x0
04048h	0404Bh	"XON Received Count—XONRXC [0:3] (0x4048; RC)" on page 1421	0x0
0404Ch	0404Fh	"XON Transmitted Count—XONTXC [0:3] (0x404C; RC)" on page 1422	0x0
04050h	04053h	"XOFF Received Count—XOFFRXC [0:3] (0x4050; RC)" on page 1422	0x0
04054h	04057h	"XOFF Transmitted Count—XOFFTXC [0:3] (0x4054; RC)" on page 1423	0x0
04058h	0405Bh	"FC Received Unsupported Count—FCRUC [0:3] (0x4058; RC)" on page 1423	0x0
0405Ch	0405Fh	"Packets Received [64 Bytes] Count—PRC64 [0:3] (0x405C; RC)" on page 1424	0x0
04060h	04063h	"Packets Received [65-127 Bytes] Count—PRC127 [0:3] (0x4060; RC)" on page 1424	0
04064h	04067h	"Packets Received [128-255 Bytes] Count—PRC255 [0:3] (0x4064; RC)" on page 1425	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 8 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
04068h	0406Bh	"Packets Received [256-511 Bytes] Count—PRC511 [0:3] (0x4068; RC)" on page 1426	0x0
0406Ch	0406Fh	"Packets Received [512-1023 Bytes] Count—PRC1023 [0:3] (0x406C; RC)" on page 1426	0x0
04070h	04073h	"Packets Received [1024 to Max Bytes] Count—PRC1522 [0:3] (0x4070; RC)" on page 1427	0x0
04074h	04077h	"Good Packets Received Count—GPRC [0:3] (0x4074; RC)" on page 1427	0x0
04078h	0407Bh	"Broadcast Packets Received Count - BPRC [0:3] (0x4078; RC)" on page 1428	0x0
0407Ch	0407Fh	"Multicast Packets Received Count—MPRC [0:3] (0x407C; RC)" on page 1428	0x0
04080h	04083h	"Good Packets Transmitted Count—GPTC [0:3] (0x4080; RC)" on page 1429	0h
04088h	0408Bh	"Good Octets Received Count—GORCL [0:3] (0x4088; RC)" on page 1429	0x0
0408Ch	0408Fh	"Good Octets Received Count—GORCH [0:3] (0x408C; RC)" on page 1430	0x0
04090h	04093h	"Good Octets Transmitted Count—GOTCL [0:3] (0x4090; RC)" on page 1430	0x0
04094h	04097h	"Good Octets Transmitted Count—GOTCH [0:3] (4094; RC)" on page 1431	0x0
040A0h	040A3h	"Receive No Buffers Count—RNBC [0:3] (0x40A0; RC)" on page 1431	0x0
040A4h	040A7h	"Receive Undersize Count—RUC [0:3] (0x40A4; RC)" on page 1432	0x0
040A8h	040ABh	"Receive Fragment Count—RFC [0:3] (0x40A8; RC)" on page 1432	0x0
040ACh	040AFh	"Receive Oversize Count—ROC [0:3] (0x40AC; RC)" on page 1433	0x0
040B0h	040B3h	"Receive Jabber Count—RJC [0:3] (0x40B0; RC)" on page 1433	0x0
040B4h	040B7h	"Management Packets Received Count—MNGPRC [0:3] (0x40B4; RC)" on page 1434	0x0
040B8h	040BBh	"Management Packets Dropped Count—MPDC [0:3] (0x40B8; RC)" on page 1434	0x0
040BCh	040BFh	"Management Packets Transmitted Count—MNGPTC [0:3] (0x40BC; RC)" on page 1435	0x0
040C0h	040C3h	"Total Octets Received—TORL [0:3] (0x40C0; RC)" on page 1435	0x0
040C4h	040C7h	"Total Octets Received—TORH [0:3] (0x40C4; RC)" on page 1436	0x0
040C8h	040CBh	"Total Octets Transmitted—TOTL [0:3] (0x40C8; RC)" on page 1436	0x0
040CCh	040CFh	"Total Octets Transmitted—TOTH [0:3] (0x40CC; RC)" on page 1437	0x0
040D0h	040D3h	"Total Packets Received—TPR [0:3] (0x40D0; RC)" on page 1437	0x0
040D4h	040D7h	"Total Packets Transmitted—TPT [0:3] (0x40D4; RC" on page 1438	0x0
040D8h	040DBh	"Packets Transmitted [64 Bytes] Count—PTC64 [0:3] (0x40D8; RC)" on page 1438	0x0
040DCh	040DFh	"Packets Transmitted [65-127 Bytes] Count—PTC127 [0:3] (0x40DC; RC)" on page 1439	0x0
040E0h	040E3h	"Packets Transmitted [128-255 Bytes] Count—PTC255 [0:3] (0x40E0; RC)" on page 1439	0x0
040E4h	040E7h	"Packets Transmitted [256-511 Bytes] Count—PTC511 [0:3] (0x40E4; RC)" on page 1440	0x0
040E8h	040EBh	"Packets Transmitted [512-1023 Bytes] Count—PTC1023 [0:3] (0x40E8; RC)" on page 1440	0x0
040ECh	040EFh	"Packets Transmitted [1024 Bytes or Greater] Count—PTC1522 [0:3] (0x40EC; RC)" on page 1441	0x0
040F0h	040F3h	"Multicast Packets Transmitted Count—MPTC [0:3] (0x40F0; RC)" on page 1441	0x0
040F4h	040F7h	"Broadcast Packets Transmitted Count—BPTC [0:3] (0x40F4; RC)" on page 1442	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 9 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
040F8h	040FBh	"TCP Segmentation Context Transmitted Count—TSCTC [0:3] (0x40F8; RC)" on page 1442	0x0
04100h	04103h	"Interrupt Assertion Count—IAC [0:3] (0x4100; RC)" on page 1443	0x0
04104h	04107h	"Rx Packets to Host Count—RPTHC [0:3] (0x4104; RC)" on page 1443	0x0
04108h	0410Bh	"Debug Counter 1—DBGC1 [0:3] (0x4108; RC)" on page 1443	0x0
0410Ch	0410Fh	"Debug Counter 2—DBGC2 [0:3] (0x410C; RC)" on page 1444	0x0
04110h	04113h	"Debug Counter 3—DBGC3 [0:3] (0x4110; RC)" on page 1445	0x0
0411Ch	0411Fh	"Debug Counter 4—DBGC4 [0:3] (0x411C; RC)" on page 1445	0x0
04118h	0411Bh	"Host Good Packets Transmitted Count—HGPTC [0:3] (0x4118; RC)" on page 1446	0x0
04120h	04123h	"Receive Descriptor Minimum Threshold Count—RXDMTC [0:3] (0x4120; RC)" on page 1446	0x0
04128h	0412Bh	"Host Good Octets Received Count—HGORCL [0:3] (0x4128; RC)" on page 1447	0x0
0412Ch	0412Fh	"Host Good Octets Received Count—HGORCH [0:3] (0x412C; RC)" on page 1447	0x0
04130h	04133h	"Host Good Octets Transmitted Count—HGOTCL [0:3] (0x4130; RC)" on page 1448	0x0
04134h	04137h	"Host Good Octets Transmitted Count - HGOTCH [0:3] (0x4134; RC)" on page 1448	0x0
04138h	0413Bh	"Length Error Count—LENERRS [0:3] (0x4138; RC)" on page 1449	0x0
04228h	0422Bh	"SerDes/SGMII/KX Code Violation Packet Count—SCVPC [0:3] (0x4228; RW)" on page 1449	0x0
041A4h	041A7h	"Switch Drop Packet Count—SDPC [0:3] (0x41A4; RC)" on page 1450	0x0
10010h at 0x100	10013h at 0x100	"Per Queue Good Packets Received Count—VFGPRC [0:3][0:7](0x10010 + n*0x100 [n=0...7]; RO)" on page 1450	0x0
10014h at 0x100	10017h at 0x100	"Per Queue Good Packets Transmitted Count—VFGPTC [0:3][0:7](0x10014 + n*0x100 [n=0...7]; RO)" on page 1451	0x0
10018h at 0x100	1001Bh at 0x100	"Per Queue Good Octets Received Count—VFGORC [0:3][0:7] (0x10018 + n*0x100 [n=0...7]; RO)" on page 1452	0x0
10034h at 0x100	10037h at 0x100	"Per Queue Good Octets Transmitted Count—VFGOTC [0:3][0:7] (0x10034 + n*0x100 [n=0...7]; RO)" on page 1453	0x0
10038h at 0x100	1003Bh at 0x100	"Per Queue Multicast Packets Received Count—VFMPRC [0:3][0:7] (0x10038 + n*0x100 [n=0...7]; RO)" on page 1453	0x0
04140h	04143h	"BMC Management Packets Dropped Count—BMPDC [0:3] (0x4140; RC)" on page 1454	0x0
4144h	4147h	"BMC Management Packets Transmitted Count—BMNGPTC [0:3] (0x4144; RC)" on page 1454	0x0
413Ch	413Fh	"BMC Management Packets Received Count—BMNGPRC [0:3] (0x413C; RC)" on page 1455	0x0
4400h	4403h	"BMC Total Unicast Packets Received—BUPRC [0:3] (0x4400; RC)" on page 1455	0x0
4404h	4407h	"BMC Total Multicast Packets Received—BMPRC [0:3] (0x4404; RC)" on page 1456	0x0
4408h	440Bh	"BMC Total Broadcast Packets Received—BBPRC [0:3] (0x4408; RC)" on page 1456	0x0
440Ch	440Fh	"BMC Total Unicast Packets Transmitted—BUPTC [0:3] (0x440C; RC)" on page 1456	0x0
4410h	4413h	"BMC Total Multicast Packets Transmitted—BMPTC [0:3] (0x4410; RC)" on page 1457	0x0
4414h	4417h	"BMC Total Broadcast Packets Transmitted—BBPTC [0:3] (0x4414; RC)" on page 1457	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 10 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
4418h	441Bh	"BMC FCS Receive Errors—BCRCERRS [0:3] (0x4418; RC)" on page 1458	0x0
441Ch	441Fh	"BMC Alignment Errors—BALGNERRC [0:3] (0x441C; RC)" on page 1458	0x0
4420h	4423h	"BMC Pause XON Frames Received—BXONRXC [0:3] (0x4420; RC)" on page 1458	0x0
4424h	4427h	"BMC Pause XOFF Frames Received—BXOFFRXC [0:3] (0x4424; RC)" on page 1459	0x0
4428h	442Bh	"BMC Pause XON Frames Transmitted—BXONTXC [0:3] (0x4428; RC)" on page 1459	0x0
442Ch	442Fh	"BMC Pause XOFF Frames Transmitted—BXOFFTXC [0:3] (0x442C; RC)" on page 1460	0x0
4430h	4433h	"BMC Single Collision Transmit Frames—BSCC [0:3] (0x4430; RC)" on page 1460	0x0
4434h	4437h	"BMC Multiple Collision Transmit Frames—BMCC [0:3] (0x4434; RC)" on page 1460	0x0
5800h	5803h	"Wakeup Control Register—WUC [0:3] (0x5800; R/W)" on page 1461	0x0
5808h	580Bh	"Wakeup Filter Control Register—WUFC [0:3] (0x5808; R/W)" on page 1462	0x0
5810h	5813h	"Wakeup Status Register—WUS [0:3] (0x5810; R/W1C)" on page 1463	0x0
5900h	5903h	"Wakeup Packet Length—WUPL [0:3] (0x5900; RO)" on page 1464	0x0
5A00h at 4	5A03h at 4	"Wakeup Packet Memory—WUPM [0:3][0:31] (0x5A00 + 4*n [n=0...31]; RO)" on page 1464	0x0
5838h	583Bh	"IP Address Valid—IPAV [0:3] (0x5838; R/W)" on page 1465	0x0
5840h at 8	5843h at 8	"IPv4 Address Table—IP4AT [0:3][0:3] (0x5840 + 8*n [n=0...3]; RW)" on page 1465	0x0
5880h at 4	5883h at 4	"IPv6 Address Table—IP6AT [0:3][0:3] (0x5880 + 4*n [n=0...3]; RW)" on page 1466	0x0
9000h at 100h at /10h	9003h at 100h at /10h	"Flex Filter Even Data Register Fields—FEDR [0:3][0:3][0:15] (0x9000 +16*n[0..15]; RW)" on page 1467	0x0
9004h at 100h at /10h	9007h at 100h at /10h	"Flex Filter Odd Data Register Fields—FODR [0:3][0:3][0:15] (0x9000 +16*n[0..15];RW)" on page 1468	0x0
9008h at 100h at /10h	900Ch at 100h at /10h	"Flex Filter Mask Field Register—FMFR [0:3][0:3][0:15] (0x9008 +16*n[0..15];RW)" on page 1468	0x0
90FCh at 100h	90FFh at 100h	"Flex Filter Queueing Field—FQFR [0:3][0:3][0:15] (0x90FC + 16*n[n=0..15] ;RW)" on page 1469	0x0
9A00h at 100h at /10h	9A03h at 100h at /10h	"Flex Filter Even Data Register Extended—FHFT_EXT_FEDR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)" on page 1470	0x0
9A04h at 100h at /10h	9A07h at 100h at /10h	"Flex Filter Odd Data Register Extended—FHFT_EXT_FODR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)" on page 1471	0x0
9A08h at 100h at /10h	9A0Ch at 100h at /10h	"Flex Filter Mask Field Extended—FHFT_EXT_FMFR [0:3][0:3][0:15] (0x9A08 +16*n[0..15]; RW)" on page 1471	0x0
9AFCh at 100h	9AFFh at 100h	"Flex Filter Queueing Extended—FHFT_EXT_FQFR [0:3][0:3] (0x9AFC; RW)" on page 1472	0x0
5010h at 4	5013h at 4	"Management VLAN TAG Value—MAVTV [0:3] [0:7] (0x5010 +4*n [n=0...7]; RW)" on page 1473	0x0
5030h at 4	5033h at 4	"Management Flex UDP/TCP Ports—MFUTP [0:3][0:3] (0x5030 + 4*n [n=0...3]; RW)" on page 1473	0x0
5060h at 4	5063h at 4	"Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW)" on page 1474	0x0
5820h	5823h	"Management Control Register—MANC [0:3] (0x5820; RW)" on page 1475	0x0
5864h	5867h	"Management Only Traffic Register—MNGONLY [0:3] (0x5864; RW)" on page 1476	0x0

Table 28-5. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR0[03]B:0:1+Index 1 Registers (Sheet 11 of 11)

Offset Start	Offset End	Register ID - Description	Default Value
5890h at 4	5893h at 4	"Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW)" on page 1476	0x0
5930h at 4	5933h at 4	"Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW)" on page 1478	0x20000000
58B0h at 4	58B3h at 4	"Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW)" on page 1481	0x0
5910h at 8	5913h at 8	"Manageability MAC Address Low—MMAL [0:3][0:1] (0x5910 + 8*n [n= 0...1]; RW)" on page 1481	0x0
5914h at 8	5917h at 8	"Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0...1]; RW)" on page 1482	0x0
1084h	1087h	"Parity and ECC Error Indication—PEIND [0:3] (0x1084; RC)" on page 1484	0x0
1088h	108Bh	"Parity and ECC Indication Mask—PEINDM [0:3] (0x1088; RW)" on page 1485	0xF
3510h	3513h	"DMA Transmit Descriptor Parity Status—DTPARS [0:3] (0x3510; RW1C)" on page 1485	0x0
3514h	3517h	"DMA Receive Descriptor Parity Status—DRPARS [0:3] (0x3514; RW1C)" on page 1486	0x0
3518h	351Bh	"Dhost Parity Status—DDPARS [0:3] (0x3518; RW1C)" on page 1486	0x0
345Ch	345Fh	"Tx Packet Buffer ECC Status—TPBECCSTS [0:3] (0x345C; RW)" on page 1487	0x00010000
5F54h	5F57h	"LAN Port Parity Error Control Register—LANPERRCTL [0:3] (0x5F54; RW)" on page 1487	0x0
5F58h	5F5Bh	"LAN Port Parity Error Status Register—LANPERRSTS [0:3] (0x5F58; RO)" on page 1488	0x0

Table 28-6. Bus B, Device 0, Function 1 + Index 1: Summary of PCI GBEPCIBAR3[03]B:0:1+Index 1 Registers

Offset Start	Offset End	Register ID - Description	Default Value
0000h at 0x10	0003h at 0x10	"MSIX Table Entry Lower Address—MSIXTADD [0:3][0:9] (BAR3: 0x0000 + 0x10*n [n=0...9]; R/W)" on page 1321	0x0
0004h at 0x10	0007h at 0x10	"MSIX Table Entry Upper Address—MSIXTUADD [0:3][0:9] (BAR3: 0x0004 + 0x10*n [n=0...9]; R/W)" on page 1322	0x0
0008h at 0x10	000Bh at 0x10	"MSIX Table Entry Message—MSIXTMSG [0:3][0:9] (BAR3: 0x0008 + 0x10*n [n=0...9]; R/W)" on page 1322	0x0
000Ch at 0x10	000Fh at 0x10	"MSIX Table Entry Vector Control—MSIXTVCTRL [0:3][0:9] (BAR3: 0x000C + 0x10*n [n=0...9]; R/W)" on page 1323	0x00000001
2000h	2003h	"MSIXPBA Bit Description—MSIXPBA [0:3] (BAR3: 0x2000; RO)" on page 1323	0x0

28.1.4 Alias Addresses

Certain registers maintain an alias address designed for backward compatibility with software written previously. For these registers, the alias address is shown in Table 28-7. Those registers can be accessed by software at either the new offset or the alias offset. It is recommended that software that is written solely for the PCH use the new address offset.

Table 28-7. Register Summary

Offset	Alias Offset	Abbreviation	Name	RW
General				
Interrupts				
0x1510	0x00E0	IAM	Interrupt Acknowledge Auto Mask	RW
Receive				
0xC000	0x0110, 0x2800	RDBAL[0]	RX Descriptor Base Low queue 0	RW
0xC004	0x0114, 0x2804	RDBAH[0]	RX Descriptor Base High queue 0	RW
0xC008	0x0118, 0x2808	RDLEN[0]	RX Descriptor Ring Length queue 0	RW
0xC010	0x0120, 0x2810	RDH[0]	RX Descriptor Head queue 0	RO
0xC018	0x0128, 0x2818	RDT[0]	RX Descriptor Tail queue 0	RW
0xC028	0x02828	RXDCTL[0]	Receive Descriptor Control queue 0	RW
0xC040 + 0x40 * (n-1)	0x2900+ 0x100 * (n-1)	RDBAL[1 - 3]	RX Descriptor Base Low queue 1 - 3	RW
0xC044 + 0x40 * (n-1)	0x2904 + 0x100 * (n-1)	RDBAH[1 - 3]	RX Descriptor Base High queue 1 - 3	RW
0xC048 + 0x40 * (n-1)	0x2908 + 0x100 * (n-1)	RDLEN[1 - 3]	RX Descriptor Ring Length queue 1 - 3	RW
0xC050 + 0x40 * (n-1)	0x2910 + 0x100 * (n-1)	RDH[1 - 3]	RX Descriptor Head queue 1 - 3	RO
0xC058 + 0x40 * (n-1)	0x2918 + 0x100 * (n-1)	RDT[1 - 3]	RX Descriptor Tail queue 1 - 3	RW
0xC068 + 0x40 * (n-1)	0x2928 + 0x100 * (n-1)	RXDCTL[1 - 3]	Receive Descriptor Control queue 1 - 3	RW
0xC100 + 0x40 * (n- 4)	N/A	RDBAL[4-7]	RX Descriptor Base Low queue 4 - 7	RW
0x5200- 0x53FC	0x0200- 0x03FC	MTA[127:0]	Multicast Table Array (n)	RW
0x5400 + 8*n	0x0040 + 8*n	RAL[0-15]	Receive Address Low (15:0)	RW
0x5404 + 8 *n	0x0044 + 8 *n	RAH[0-15]	Receive Address High (15:0)	RW
0x5600-0x57FC	0x0600-0x07 FC	VFTA[127:0]	VLAN Filter Table Array (n)	RW
Transmit				
0xE000	0x0420, 0x3800	TDBAL[0]	TX Descriptor Base Low 0	RW
0xE004	0x0424, 0x3804	TDBAH[0]	TX Descriptor Base High 0	RW
0xE008	0x0428, 0x3808	TDLEN[0]	TX Descriptor Ring Length 0	RW

Table 28-7. Register Summary (Continued)

Offset	Alias Offset	Abbreviation	Name	RW
0xE010	0x0430, 0x3810	TDH[0]	TX Descriptor Head 0	RO
0xE018	0x0438, 0x3818	TDT[0]	TX Descriptor Tail 0	RW
0xE038	0x3838	TDWBAL[0]	Transmit Descriptor WB Address Low queue 0	RW
0xE03C	0x383C	TDWBAH[0]	Transmit Descriptor WB Address High queue 0	RW
0xE040 + 0x40 * (n-1)	0x3900 + 0x100 * (n-1)	TDBAL[1-3]	TX Descriptor Base Low queue 1 - 3	RW
0xE044 + 0x40 * (n-1)	0x3904 + 0x100 * (n-1)	TDBAH[1-3]	TX Descriptor Base High queue 1 - 3	RW
0xE048 + 0x40 * (n-1)	0x3908 + 0x100 * (n-1)	TDLEN[1-3]	TX Descriptor Ring Length queue 1 - 3	RW
0xE050 + 0x40 * (n-1)	0x3910 + 0x100 * (n-1)	TDH[1-3]	TX Descriptor Head queue 1 - 3	RO
0xE058 + 0x40 * (n-1)	0x3918 + 0x100 * (n-1)	TDT[1-3]	TX Descriptor Tail queue 1 - 3	RW
0xE078 + 0x40 * (n-1)	0x3938 + 0x100 * (n-1)	TDWBAL[1-3]	Transmit Descriptor WB Address Low queue 1 - 3	RW
0xE07C + 0x40 * (n-1)	0x393C + 0x100 * (n-1)	TDWBAH[1-3]	Transmit Descriptor WB Address High queue 1 - 3	RW
0xE1B8 + 0x40 * (n - 4)	N/A	TDWBAL[4 -]	Transmit Descriptor WB Address Low queue 4 -	RW
Filters				
VMDq Statistics				
Statistics				

28.1.4.1 MSI-X BAR Register Summary

Table 28-8. MSI-X Register Summary

Category	Offset	Abbreviation	Name	RW	Page
MSI-X Table	0x0000 + n*0x10 [n=0...9]	MSIXTADD	MSI–X Table Entry Lower Address	RW	page 1321
MSI-X Table	0x0004 + n*0x10 [n=0...9]	MSIXTUADD	MSI–X Table Entry Upper Address	RW	page 1322
MSI-X Table	0x0008 + n*0x10 [n=0...9]	MSIXTMSG	MSI–X Table Entry Message	R/W	page 1322
MSI-X Table	0x000C + n*0x10 [n=0...9]	MSIXTVCTRL	MSI–X Table Entry Vector Control	R/W	page 1323
MSI-X Table	0x02000	MSIXPBA	MSIXPBA Bit Description	RO	page 1323

28.1.4.2 Device Control Register—CTRL [0:3] (0x00000; R/W)

This register, as well as the Extended Device Control register (CTRL_EXT), controls the major operational modes for the device. While software write to this register to control device settings, several bits (such as FD and SPEED) can be overridden depending on other bit settings and the resultant link configuration determined by the PHY's Auto-Negotiation resolution. See Section 23.4.6 for details on the setup of these registers in the different link modes.

Note: This register is also aliased at address 0x0004.

Table 28-9. Device Control Register—CTRL [0:3] (0x00000; R/W) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0000h **Offset End:** 0003h	
Size: 32 bit	**Default:** 0x08100201			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	PHY_RST	PHY Reset For SW initiated PHY reset the driver sets this bit, resets the external PHY through MDIO interface and then clears the PHY_RST bit.		0h	R/W
30	VME	VLAN Mode Enable When set to 1b, VLAN information is stripped from all received 802.1Q packets.		0h	R/W
29	DEV_RST (SC)	Device Reset This bit performs a reset of the entire controller device, resulting in a state nearly approximating the state following a power-up reset or internal PCIe* reset, except for system PCI configuration. 0b = Normal 1b = Reset This bit is self clearing. Notes: 1. Assertion of DEV_RST generates an interrupt on all ports via the ICR.DRSTA interrupt bit. 2. Device Reset (CTRL.DEV_RST) can be used to globally reset the entire component if the DEV_RST_EN bit in Initialization Control 4 EEPROM word is set. 3. Assertion of DEV_RST sets on all ports the *STATUS.DEV_RST_SET* bit. For additional information see Section 28.2.1.2.		0h	R/W
28	TFCE	Transmit Flow Control Enable When set, indicates that the Controller transmits flow control packets (XON and XOFF frames) based on the receiver fullness. If Auto-Negotiation is enabled, this bit is set to the negotiated duplex value. In SerDes mode the resolution is done by the hardware. In SGMII or 1000BASE-KX modes software should do it.		0h	R/W
27	RFCE	Receive Flow Control Enable When set, indicates that the Controller responds to the reception of flow control packets. If Auto-Negotiation is enabled, this bit is set to the negotiated flow control value. In SerDes mode the resolution is done by the hardware. In SGMII or 1000BASE-KX modes it should be done by the software.		1h	R/W
26	RST	Port Software Reset This bit performs reset to the respective port, resulting in a state nearly approximating the state following a power-up reset or internal PCIe* reset, except for system PCI configuration and logic used by all ports. 0b = Normal 1b = Reset This bit is self clearing and is referred to as software reset or global reset.		0h	R/W
25 :24	Reserved	Reserved.			
23	SDP1_IODIR	SDP1 Pin Direction Controls whether software-controllable pin SDP1 is configured as an input or output (0b = input, 1b = output). Initial value is EEPROM-configurable. This bit is not affected by software or system reset, only by initial power-on or direct software writes.		0h[1]	R/W

Table 28-9. Device Control Register—CTRL [0:3] (0x00000; R/W) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0000h **Offset End:** 0003h	
Size: 32 bit	**Default:** 0x08100201			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
22	SDP0_IODIR	SDP0 Pin Direction Controls whether software-controllable pin SDP0 is configured as an input or output (0b = input, 1b = output). Initial value is EEPROM-configurable. This bit is not affected by software or system reset, only by initial power-on or direct software writes.		0h[1]	R/W
21	SDP0_WDE	SDP0 used for Watchdog indication When set, SDP0 is used as a watchdog indication, and the SDP0_DATA bit indicates the polarity of the watchdog indication. In this mode, S*DP0_IODIR* must be set to an output.		0h[1]	R/W
20	ADVD3WUC	D3Cold Wake up Capability Advertisement Enable When set, D3Cold wake up capability is advertised based on whether AUX_PWR advertises presence of auxiliary power (yes if AUX_PWR is indicated, no otherwise). When 0b, D3Cold wake up capability is not advertised even if AUX_PWR presence is indicated. The initial value is EEPROM configurable. If full 1Gb/sec. operation in D3 state is desired but the system's power requirements in this mode would exceed the D3Cold Wake. up-Enabled specification limit (375mA at 3.3V), this bit can be used to prevent the capability from being advertised to the system. See bit 2 of Software Defined Pins Control (LAN Base Address + Offset 0x20) in EEPROM.		[1]	
19	SDP1 DATA (RWS)	SDP1 Data Value Used to read or write the value of software-controlled IO pin SDP1. If SDP1 is configured as an output (SDP1_IODIR = 1b), this bit controls the value driven on the pin (initial value EEPROM-configurable). If SDP1 is configured as an input, reads return the current value of the pin.		0h[1]	R/W
18	SDP0 DATA (RWS)	SDP0 Data Value Used to read or write the value of software-controlled IO pin SDP0. If SDP0 is configured as an output (*SDP0_IODIR* = 1b), this bit controls the value driven on the pin (initial value EEPROM-configurable). If SDP0 is configured as an input, reads return the current value of the pin. When the SDP0_WDE bit is set, this field indicates the polarity of the watchdog indication.		0h[1]	R/W
17	SDP1_GPIEN	General Purpose Interrupt Detection Enable for SDP1 If software-controlled IO pin SDP1 is configured as an input, this bit (when 1b) enables the use for GPI interrupt detection.		0h	R/W
16	SDP0_GPIEN	General Purpose Interrupt Detection Enable for SDP0 If software-controlled IO pin SDP0 is configured as an input, this bit (when 1b) enables the use for GPI interrupt detection.		0h	R/W
15 :13	Reserved	Reserved Write 0, ignore on read.			
12	FRCDPLX	Force Duplex When set to 1b, software can override the duplex indication from the PHY. Otherwise, in 10/100/1000Base-T link mode, the duplex setting is sampled from the PHY into the MAC on the asserting edge of the PHY LINK signal. When asserted, the *CTRL.FD* bit sets duplex.		0h	R/W

Table 28-9. Device Control Register—CTRL [0:3] (0x00000; R/W) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0000h **Offset End:** 0003h	
Size: 32 bit	**Default:** 0x08100201			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11	FRCSPD	Force Speed This bit is set when software needs to manually configure the MAC speed settings according to the SPEED bits. MAC and PHY must resolve to the same speed configuration or software must manually set the PHY to the same speed as the MAC. Software must clear this bit to enable the PHY or ASD function to control the MAC speed setting. This bit is superseded by the *CTRL_EXT.SPD_BYPS* bit which has a similar function.		0h[1]	R/W
10	Reserved	Reserved. Write as 0b to ensure future compatibility.			
09 :08	SPEED	Speed selection. These bits determine the speed configuration and are written by software after reading the PHY configuration through the MDIO interface. These signals are ignored when Auto-Speed Detection is enabled. 00b = 10 Mb/s. 01b = 100 Mb/s. 10b = 1000 Mb/s. 11b = not used.		2h	R/W
07	ILOS	Invert Loss-of-Signal (LOS/LINK) Signal Bit controls the polarity of the SRDS_[n]_SIG_DET signal or internal Link up signal depending on the *CONNSW.ENRGSRC* bit. 0b = Do not invert (active high input signal). 1b = Invert signal (active low input signal). Should be set to zero when working in 1000BASE-KX mode. ***Note:*** ILOS bit value is updated to Initial Value only after PCIe* reset.		0h[1]	R/W
06	SLU	Set Link Up. See Section 22.5.4 for more information about Auto-Negotiation and link configuration in the various modes. The "Set Link Up" is normally initialized to 0. However, if the *APM Enable* bit is set in the EEPROM then it is initialized to 1b. In SerDes and 1000Base-KX modes Link up can be forced by setting this bit as described in Section 22.5.4.1.4.		0h[1]	R/W
05 :03	Reserved	Reserved. Write 0 ignore on read.			
02	GIO Master Disable	When set to 1b, the function of this bit blocks new master requests including manageability requests. If no master requests are pending by this function, the *STATUS.GIO Master Enable Status* bit is set. See Section 25.2.3.2 for further information.		0h	R/W
01	Reserved	This bit is reserved and should be set to 0b for future compatibility.			
00	FD	Full-Duplex Controls the MAC duplex setting when explicitly set by software. 0b = half duplex. 1b = full duplex.		1h[1]	R/W

1. These bits are loaded from EEPROM.

28.1.4.3 Device Status Register—STATUS [0:3] (0x0008; R)

Table 28-10. Device Status Register— STATUS[0:3] (0x0008; R) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 00008h **Offset End:** 0000Bh	
Size: 32 bit	**Default:** 0x80080400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	MAC clock gating Enable	MAC clock gating Enable bit loaded from the EEPROM- indicates the device support gating of the MAC clock.		1h[1]	R
30 :21	Reserved	Reserved			
20	DEV_RST_SET (R/W1C)	Device Reset Set When set indicates that a device reset (*CTRL.DEV_RST*) was initiated by one of the software drivers. Note: Bit cleared by write 1.		0h	R
19	GIO Master Enable Status	Cleared by the Controller when the *CTRL.GIO Master Disable* bit is set and no master requests are pending by this function and is set otherwise. Indicates that no master requests are issued by this function as long as the *CTRL.GIO Master Disable* bit is set.		1h	R
18 :10	Reserved	Reserved			
09 :08	ASDV	Auto-Speed Detection Value Speed result sensed by the Controller's MAC auto-detection function. These bits are provided for diagnostics purposes only. The ASD calculation can be initiated by software writing a logic 1b to the *CTRL_EXT.ASDCHK* bit. The resultant speed detection is reflected in these bits. See Section 28.1.4.4 for details.		X	R
07 :05	Reserved	Reserved			
04	TXOFF	Transmission Paused This bit indicates the state of the transmit function when symmetrical flow control has been enabled and negotiated with the link partner. This bit is set to 1b when transmission is paused due to the reception of an XOFF frame. It is cleared (0b) upon expiration of the pause timer or the receipt of an XON frame.		X	R

Table 28-10. Device Status Register— STATUS[0:3] (0x0008; R) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 00008h **Offset End:** 0000Bh	
Size: 32 bit	**Default:** 0x80080400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
03 :02	LAN ID	LAN ID Provides software a mechanism to determine the LAN identifier for the MAC. 00b = LAN 0. 01b = LAN 1. 10b = LAN 2. 11b = LAN 3.		X	R
01	LU	Link Up. 0 = no link established; 1 = link established. For this to be valid, the Set Link Up bit of the Device Control Register (CTRL.SLU) must be set. Link up provides a useful indication of whether something is attached to the port. Successful negotiation of features/link parameters results in link activity. The link startup process (and consequently the duration for this activity after reset) can be several 100's of ms. When the SerDes, SGMII or 1000BASE-KX interface is used, this indicates loss-of-signal; if Auto-Negotiation is also enabled, this can also indicate successful Auto-Negotiation. See Section 22.5.4 for more details. Note: In SerDes mode bit is always 0.		X	R
00	FD	Full Duplex. 0 = Half duplex (HD). 1= Full duplex (FD). Reflects duplex setting of the MAC and/or link. FD reflects the actual MAC duplex configuration. This normally reflects the duplex setting for the entire link, as it normally reflects the duplex configuration negotiated between the PHY and link partner (copper link) or MAC and link partner (fiber link).		X	R

1. If the signature bits of the EEPROM's *Initialization Control Word 1* match (01b), this bit is read from the EEPROM.

28.1.4.4 Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W)

This register provides extended control of the Controller's functionality beyond that provided by the Device Control register (*CTRL*).

Table 28-11. Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 00018h **Offset End:** 0001Bh	
Size: 32 bit	**Default:** 0x00500000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved	Reserved Write 0, Ignore on read.			
28	DRV_LOAD	Driver Loaded This bit should be set by the driver after it is loaded. This bit should be cleared when the driver unloads or after a PCIe* soft reset. The MNG controller reads this bit to indicate to the manageability controller (BMC) that the driver has loaded.		0h	R/W
26	Extended VLAN	Extended VLAN When set, all incoming Rx packets are expected to have at least one VLAN with the Ether type as defined in VET.EXT_VET that should be ignored. The packets can have a second VLAN that should be used for all filtering purposes. All Tx packets are expected to have at least one VLAN added to them by the host. In the case of an additional VLAN request (VLE - VLAN Enable is set in transmit descriptor) the second VLAN is added after the first VLAN is added by the host. This bit is reset only by a power up reset or by an EEPROM full auto load and should only be changed while Tx and Rx processes are stopped.		0h[1]	R/W
25	I2C Enabled	Enable I2C This bit enables the SFPx_I2C pins that can be used to access external SFP modules or an external 1000BASE-T PHY via the MDIO interface. If cleared, the SFPx_I2C pads are isolated and accesses to the SFPx_I2C pins through the *I2CCMD* register or the *MDIC* register are ignored.		0h[1]	R/W
24	Reserved	Reserved. Write 0, ignore on read.			
23 :22	LINK_MODE	Link Mode Controls interface on the link. 00b = Default Note: This is the lowest power mode. This is default to keep MAC in lowest power mode by default. 01b = 1000BASE-KX. 10b = SGMII. 11b = SerDes interface.		0x00	R/W
21	Reserved	Reserved. Should be set to 0b.			
20	Reserved	Reserved			
19	Dynamic MAC Clock Gating	When set, enables Dynamic MAC Clock Gating.		0h[1]	R/W
18	SerDes Low Power Enable	When set, allows the SerDes to enter a low power state when the function is in Dr state.		0h[1]	R/W
17	RO_DIS	Relaxed Ordering Disabled When set to 1b, the Controller does not request any relaxed ordering transactions on the PCIe* interface regardless of the state of bit 4 in the *PCIe* Device Control* register. When this bit is cleared and bit 4 of the *PCIe* Device Control* register is set, the Controller requests relaxed ordering transactions as specified by registers *RXCTL* and *TXCT*L (per queue and per flow).		0h	R/W

Table 28-11. Extended Device Control Register—CTRL_EXT [0:3] (0x0018; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 00018h **Offset End:** 0001Bh	
Size: 32 bit	**Default:** 0x00500000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
16	NS_DIS	No Snoop Disable When set to 1b, the Controller does not set the no snoop attribute in any PCIe* packet, independent of PCIe* configuration and the setting of individual no snoop enable bits. When set to 0b, behavior of no snoop is determined by PCIe* configuration and the setting of individual no snoop enable bits.		0h	R/W
15	SPD_BYPS	Speed Select Bypass When set to 1b, all speed detection mechanisms are bypassed, and the Controller is immediately set to the speed indicated by *CTRL.SPEED*. This provides a method for software to have full control of the speed settings of the Controller and when the change takes place, by overriding the hardware clock switching circuitry.		0h	R/W
14	Reserved	Reserved			
13	EE_RST	EEPROM Reset When set, initiates a reset-like event to the EEPROM function. This causes the EEPROM to be read as if a RST# assertion had occurred. All Controller functions should be disabled prior to setting this bit. This bit is self-clearing.		0h	RW1C
12	ASDCHK	ASD Check Initiates an Auto-Speed-Detection (ASD) sequence to sense the frequency of the PHY receive clock (RX_CLK). The results are reflected in STATUS.ASDV. This bit is self-clearing.		0h	RW1C
11 :00	Reserved	Reserved. Should be written as 0b to ensure future compatibility.		0h	R/W

The PCH allows up to two externally controlled interrupts. All software-definable pins, these can be mapped for use as GPI interrupt bits. Mappings are enabled by the SDPx_GPIEN bits only when these signals are also configured as inputs via SDPx_IODIR. When configured to function as external interrupt pins, a GPI interrupt is generated when the corresponding pin is sampled in an active-high state.

The bit mappings are shown in the table bellow for clarity.

Table 28-12. Mappings for SDI Pins Used as GPI

SDP Pin Used as GPI	**CTRL_EXT Field Settings**		**Resulting ICR Bit (GPI)**
	Direction	**Enable as GPI interrupt**	
1	SDP1_IODIR	SDP1_GPIEN	12
0	SDP0_IODIR	SDP0_GPIEN	11

Note: If software uses the EE_RST function and desires to retain current configuration information, the contents of the control registers should be read and stored by software. Control register values are changed by a read of the EEPROM which occurs upon assertion of the EE_RST bit.

Note: The EEPROM reset function can read configuration information out of the EEPROM which affects the configuration of PCIe* space BAR settings. The changes to the BARs are not visible unless the system reboots and the BIOS is allowed to re-map them.

- The SPD_BYPS bit performs a similar function to the CTRL.FRCSPD bit in that the Controller's speed settings are determined by the value software writes to the CRTL.SPEED bits. However, with the SPD_BYPS bit asserted, the settings in CTRL.SPEED take effect immediately rather than waiting until after the Controller's clock switching circuitry performs the change.

28.1.4.5 MDI Control Register—MDIC [0:3] (0x0020; R/W)

Software uses this register to read or write Management Data Interface (MDI) registers in an external SGMII PHY.

See Section 22.5.2.2.2 for details on usage of this register.

Table 28-13. MDI Control Register—MDIC[0:3] (0x0020; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 00020h **Offset End:** 00023h	
Size: 32 bit	**Default:** 0x10000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved.		0h	R/W
30	MDI_ERR (RWS)	Error This bit is set to 1b by hardware when it fails to complete an MDI read. Software should make sure this bit is clear (0h) before issuing an MDI read or write command. Note: bit is valid only when the Ready bit is set.		0h	R/W
29	MDI_IE	Interrupt Enable When set to 1 an Interrupt is generated at the end of an MDI cycle to indicate an end of a read or write operation to the PHY.		0h	R/W
28	R (RWS)	Ready Bit Set to 1b by the Controller at the end of the MDI transaction (for example, indication of a Read or Write completion). It should be reset to 0b by software at the same time the command is written.		1h	R/W
27 :26	OP	Opcode 01b = MDI Write 10b = MDI Read All other values are reserved.		0x0	R/W
25 :21	Reserved	Reserved.			
20 :16	REGADD	PHY Register Address: Reg. 0, 1, 2,...31		0x0	R/W
15 :00	DATA	Data In a Write command, software places the data bits and the MAC shifts them out to the PHY. In a Read command, the MAC reads these bits serially from the PHY and software can read them from this location.		X	R/W

28.1.4.6 MDC/MDIO Configuration Register—MDICNFG [0:3] (0x0E04; R/W)

This register is used to configure the MDIO connection that is accessed via the *MDIC* register. See Section 22.5.2.2.2 for details on usage of this register.

Table 28-14. MDC/MDIO Configuration Register—MDICNFG [0:3] (0x0E04; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 00E04h **Offset End:** 00E07h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Destination[1]	Destination 0b = Sideband interface via I2C interface when PHY Register access is initiated via the I2CCMD interface. 1b = The MDIO transaction is directed to the external MDIO pins. ***Note:*** • When PHY registers access is initiated via the I2CCMD interface, access is always via the external I2C Interface. In this case the destination field should always be 0.		0h	R/W
30	Com_MDIO[2]	When interfacing an external SGMII PHY bit defines if MDIO access is routed to the common MDIO port on LAN 0, to support multi port external PHYs, or to the dedicated per function MDIO port. 0b - MDIO access routed to the LAN port's MDIO interface. 1b - MDIO accesses on this LAN port routed to LAN port 0 MDIO interface		0h	R/W
29 :26	Reserved	Reserved. Write 0, ignore on read.			
25 :21	PHYADD[3]	External PHY Address		0x00 - LAN 0h 0x01 - LAN 1h 0x02 - LAN 2h 0x03 - LAN 3h	R/W
20 :00	Reserved	Reserved. Write 0, ignore on read.			

1. Destination Loaded from EEPROM Initialization Control 3 word. When external PHY supports a MDIO interface bit is 1, otherwise bit is 0.
2. Common MDIO usage configuration bit is loaded from *Initialization Control 3* EEPROM word.
3. PHYADD Loaded from *Initialization Control 4* EEPROM word to allocate per port address when using common MDIO port.

28.1.4.7 Copper/Fiber Switch Control—CONNSW [0:3] (0x0034; R/W)

Table 28-15. Copper/Fiber Switch Control—CONNSW[0:3] (0x0034; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0034h **Offset End:** 0037h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :03	Reserved	Reserved	N		
02	ENRGSRC	SerDes Energy Detect Source If set, the OMED interrupt cause is set after asserting the external signal detect pin. If cleared, the OMED interrupt cause is set after exiting from electrical idle of the SerDes receiver. This bit also defines the source of the signal detect indication used to set link up while is SerDes mode.	N	0h[1]	R/W
01 00	Reserved	Reserved	N		

1. The default value of the ENRGSRC bit in this register is defined in the *Initialization Control 3* (Offset 0x24) EEPROM word (bit 15).

28.1.4.8 VLAN Ether Type—VET [0:3] (0x0038; R/W)

This register contains the type field hardware matches against to recognize an 802.1Q (VLAN) Ethernet packet and uses when add and transmit VLAN Ethernet packets. To be compliant with the 802.3ac standard, this register should be programmed with the value 0x8100. For VLAN transmission the upper byte is first on the wire (VET[15:8]).

Table 28-16. VLAN Ether Type—VET [0:3] (0x0038; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0038h **Offset End:** 003Bh	
Size: 32 bit	**Default:** 0x81008100			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	VET EXT	External VLAN Ether Type.		0x8100	R/W
15 :00	VET	VLAN EtherType Should be programmed with 0x8100.		0x8100	RO

28.1.4.9 LED Control—LEDCTL [0:3] (0x0E00; RW)

This register controls the setup of the LEDs. See Section 26.5.1 for details of the MODE fields encoding.

Table 28-17. LED Control—LEDCTL [0:3](0x0E00; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E00h **Offset End:** E03h	
Size: 32 bit	**Default:** 0x07568302			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write as 0 ignore on read.			
07	LED0_BLINK	LED0/LINK# Blink This field specifies whether to apply blink logic to the (possibly inverted) LED control source prior to the LED output. 0b = Do not blink asserted LED output. 1b = Blink asserted LED output.		0h[1]	RW
06	LED0_IVRT	LED0/LINK# Invert This field specifies the polarity/ inversion of the LED source prior to output or blink control. 0b = Do not invert LED source. 1b = Invert LED source.		0h[1]	RW
05	GLOBAL_BLINK_MODE	Global Blink Mode This field specifies the blink mode of all the LEDs. 0b = Blink at 200 ms on and 200 ms off. 1b = Blink at 83 ms on and 83 ms off.		0h[1]	RW
04	Reserved	Reserved			
03 :00	LED0_MODE	LED0/LINK# Mode This field specifies the control source for the LED0 output. An initial value of 0010b selects LINK_UP# indication.		2h[1]	RW

1. These bits are read from the EEPROM.

28.2 Internal Packet Buffer Size Registers

The following registers define the size of the on-chip receive and transmit buffers used to receive and transmit packets. The overall available internal buffer size in the Controller for all ports is 144 KB for receive buffers and 80 KB for transmit Buffers. Disabled ports memory can be shared between active ports and sharing can be asymmetric. The default buffer size for each port is loaded from the EEPROM on initialization.

28.2.1 Detailed Register Descriptions

28.2.1.1 Internal Receive Packet Buffer Size—IRPBS [0:3](0x2404; RO)

Table 28-18. Internal Receive Packet Buffer Size—IRPBS [0:3] (0x2404; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2404h **Offset End:** 2407h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03 :00	RXPbsize[1]	Receive internal buffer size: 0x0 - 36 KB 0x1 - 72 KB 0x2 - 144 KB 0x3 - 1 KB 0x4 - 2 KB 0x5 - 4 KB 0x6 - 8 KB 0x7 - 16 KB 0x8 - 35 KB 0x9 - 70 KB 0xA - 140 KB 0xB:0xF - reserved Notes: 1. When 4 ports are active maximum buffer size can be 36 KB. When 2 ports are active maximum buffer size can be 72 KB. When only a single port is active maximum buffer size can be 144 KB. For further information see Section 26.1.3.2. 2. Values bellow 35 KB should be used for diagnostic purposes only. 3. When port is disabled for both PCIe* and Management access, the buffer size allocated to the port is 0 Bytes. Available internal memory can be used by other ports. 4. When BMC to Host traffic is enabled maximum available receive buffer for all ports is 140 KB.		0x0	RO

1. Value loaded from *Initialization Control 4* EEPROM word.

28.2.1.2 Internal Transmit Packet Buffer Size—ITPBS [0:3] (0x3404; RO)

Table 28-19. Internal Transmit Packet Buffer Size—ITPBS [0:3] (0x3404; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 3404h **Offset End:** 3407h	
Size: 32 bit	**Default:** 0x8			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03 :00	TXPbsize	Transmit internal buffer size[1] 0x0 - 20 KB 0x1 - 40 KB 0x2 - 80 KB 0x3 - 1 KB 0x4 - 2 KB 0x5 - 4 KB 0x6 - 8 KB 0x7 - 16 KB 0x8 - 19 KB 0x9 - 38 KB 0xA - 76 KB 0xB:0xF - reserved Notes: 1. When 4 ports are active maximum buffer size can be 20KB. When only 2 ports are active maximum buffer size is 40KB. When only a single port is active maximum buffer size is 80KB. 2. Values bellow 20 KB should be used for diagnostic purposes only. 3. When port is disabled for both PCIe* and management access, the buffer size allocated to the port is 0 Bytes. Available internal memory can be used by other ports.		0x8	RO

1. Value loaded from Initialization Control 4 EEPROM word.

28.3 EEPROM Registers

28.3.1 Detailed Register Descriptions

28.3.1.1 EEPROM Control and Data Register—EEC [0:3] (0x0010; R/W)

This register provides software direct access to the EEPROM. Software can control the EEPROM by successive writes to this register. Data and address information is clocked into the EEPROM by software toggling the EE_SK and EE_DI bits (0 and 2) of this register with EE_CS set to 0b. Data output from the EEPROM is latched into the EE_DO bit (bit 3) via the internal 62.5 MHz clock and can be accessed by software via reads of this register.

Table 28-20. EEPROM Control Register—EEC [0:3] (0x0010; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0010h **Offset End:** 0013h	
Size: 32 bit	**Default:** Xh			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Write 0 ignore on read.			
15	EE_BLOCKED (R/W1C)	EPROM access blocked EEPROM access blocked - Bit is set by HW when bit banging transactions are blocked due to write to read-only sections or any access to hidden area. Note: Bit is cleared by write one.		0x0	R/W
14 :11[1]	EE_SIZE (RO)	EEPROM Size This field defines the size of the EEPROM: Field Value EEPROM Size EEPROM Address Size 0111b 16 Kbytes 2 bytes 1000b 32 Kbytes 2 bytes 1011b - 1111b Reserved		2h	R/W
10	EE_ADDR_SIZE	EEPROM Address Size This field defines the address size of the EEPROM. This bit is set by the EEPROM size auto-detect mechanism. If no EEPROM is present or the signature is not valid, a 16-bit address is assumed. 0b = 8- and 9-bit. 1b = 16-bit.		0h	R/W
09	Auto_RD (RO)	EEPROM Auto Read Done When set to 1b, this bit indicates that the auto read by hardware from the EEPROM is done. This bit is also set when the EEPROM is not present or when its signature is not valid.		0h	R/W
08	EE_PRES (RO)	EEPROM Present This bit indicates that an EEPROM is present by monitoring the EE_DO input for an active-low acknowledge by the serial EEPROM during initial EEPROM scan. 1b = EEPROM present.		X	R/W
07	EE_GNT	Grant EEPROM Access When this bit is 1b the software can access the EEPROM using the SK, CS, DI, and DO bits.		0h	R/W
06	EE_REQ	Request EEPROM Access The software must write a 1b to this bit to get direct EEPROM access. It has access when EE_GNT is 1b. When the software completes the access it must write a 0b.		0h	R/W
05 :04	FWE	Reserved			
03	EE_DO (RO)	Data output bit from the EEPROM[2] The EE_DO input signal is mapped directly to this bit in the register and contains the EEPROM data output. This bit is RO from a software perspective; writes to this bit have no effect.		X	R/W

Table 28-20. EEPROM Control Register—EEC [0:3] (0x0010; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0010h **Offset End:** 0013h	
Size: 32 bit	**Default:** Xh			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	EE_DI	Data input to the EEPROM When EE_GNT = 1b, the EE_DI output signal is mapped directly to this bit. Software provides data input to the EEPROM via writes to this bit.		0h	R/W
01	EE_CS	Chip select input to the EEPROM When EE_GNT = 1b, the EE_CS output signal is mapped to the chip select of the EEPROM device. Software enables the EEPROM by writing a 1b to this bit.		0h	R/W
00	EE_SK	Clock input to the EEPROM When EE_GNT = 1b, the EE_SK output signal is mapped to this bit and provides the serial clock input to the EEPROM. Software clocks the EEPROM via toggling this bit with successive writes.		0h	R/W

1. These bits are read from the EEPROM.
2. Value depends on voltage level on EE_DO pin following initialization

28.3.1.2 EEPROM Read Register - EERD [0:3] (0x0014; RW)

This register is used by software to cause the Controller to read individual words in the EEPROM. To read a word, software writes the address to the *Read Address* field and simultaneously writes a 1b to the *Start Read* field. The Controller reads the word from the EEPROM and places it in the *Read Data* field, setting the *Read Done* field to 1b. Software can poll this register, looking for a 1b in the *Read Done* field, and then using the value in the *Read Data* field.

When this register is used to read a word from the EEPROM, that word does not influence any of the Controller's internal registers even if it is normally part of the auto-read sequence.

Table 28-21. EEPROM Read Register—EERD [0:3] (0x0014; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0014h **Offset End:** 0017h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	DATA (RO)	Read Data. Data returned from the EEPROM read.		X	RW
15 :02	ADDR	Read Address This field is written by software along with *Start Read* to indicate the word to read.		0x0	RW
01	DONE (RO)	Read Done Set to 1b when the EEPROM read completes. Set to 0b when the EEPROM read is not completed. Writes by software are ignored. Reset by setting the START bit.		0h	RW
00	START	Start Read Writing a 1b to this bit causes the EEPROM to read a (16-bit) word at the address stored in the EE_ADDR field and then storing the result in the EE_DATA field. This bit is self-clearing.		0h	RW1C

28.3.1.3 EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO)

This register reflects the values of EEPROM bits influencing the hardware that are not reflected otherwise.

Table 28-22. EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1038h **Offset End:** 103Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Deadlock Release	Indicates a deadlock condition was detected in the EEPROM and the current grant was released.		X	RO
30 :29	Reserved	Reserved			
28	LAN3 PCI Disable	Reflects bit 10 in the Software Defined Pins Control (Offset 0x20) EEPROM word controlling the disabling of LAN3 as PCIe* function.		0h	RO
27	LAN2 PCI Disable	Reflects bit 10 in the *Software Defined Pins Control* (Offset 0x20) EEPROM word controlling the disabling of LAN2 as PCIe* function.		0h	RO
26 :25	Reserved	Reserved			
24	LAN3 Disable	Reflects bit 11 in the *Software Defined Pins Control* (Offset 0x20) EEPROM word controlling the disabling of LAN3.		0h	RO

Table 28-22. EEPROM Diagnostic—EEDIAG [0:3] (0x1038; RO) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1038h **Offset End:** 103Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23	LAN2 Disable	Reflects bit 11 in the *Software Defined Pins Control* (Offset 0x20) EEPROM word controlling the disabling of LAN2.		0h	RO
22 :16	Reserved	Reserved			
15 :14	EEPROM Serial State	State of the EEPROM serial access arbitration state machine.		X	RO
13 :12	EEPROM Parallel State	State of the EEPROM parallel access arbitration state machine.		X	RO
11 :07	Reserved	Reserved			
06	PLL Shutdown Enable	Reflects bit 4 in EEPROM 0x0F controlling the PLL shutdown enable control.		0h	RO
05	Reserved	Reserved			
04	EEPROM Deadlock Release Enable	Reflects bit 5 in EEPROM word 0x0A controlling the EEPROM deadlock release enable.		0h	RO
03	LAN1 PCI Disable	Reflects bit 10 in the *Software Defined Pins Control (Offset 0x20)* EEPROM word 0x10 controlling the disabling of the LAN1 PCIe* function.		0h	RO
02	LAN1 Disable	Reflects bit 11 in the *Software Defined Pins Control (Offset 0x20)* EEPROM word controlling the disabling of LAN1.		0h	RO
01 00	Reserved	Reserved			

28.3.1.4 VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO)

This register stores the VPD parameters as parsed by the auto-load process. This register is used for debug only.

Table 28-23. VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1060h **Offset End:** 1063h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	End Tag	Offset of the End tag in VPD relative to the start of VPD (in bytes).		X	RO
22 :14	WR Tag	Offset of the Write tag in VPD relative to the start of VPD (in bytes).		X	RO

Table 28-23. VPD Diagnostic Register—VPDDIAG [0:3] (0x1060; RO) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1060h **Offset End:** 1063h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
13 :05	RD Tag	Offset of the Read tag in VPD relative to the start of VPD (in bytes).		X	RO
04 :01	Reserved	Reserved			
00	Valid	VPD structure valid		X	RO

28.3.1.5 Management—EEPROM CSR I/F

The following registers are reserved for Firmware access to the EEPROM and are not writable by the host.

28.3.1.6 Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO)

Table 28-24. Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1010h **Offset End:** 1013h	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	DONE	Transaction Done - This bit is cleared after Start Write or Start Read bit is set by the MNG and is set back again when the EEPROM write or read transaction is done.		1h	RO
30 :22	Reserved	Reserved			
21	CFG_DONE 3	Configuration cycle is done for port 3[1] – This bit indicates that the configuration cycle (configuration of SerDes, PCIe* and PLLs) is done for port 3. This bit is set to 1b to indicate configuration done, and cleared by hardware on any of the reset sources that cause initialization of the PHY. Note: Port 3 driver should not try to access the PHY for configuration before this bit is set.		0h	RO
20	CFG_DONE 2	Configuration cycle is done for port 2[1] – This bit indicates that the configuration cycle (configuration of SerDes, PCIe* and PLLs) is done for port 2. This bit is set to 1b to indicate configuration done, and cleared by hardware on any of the reset sources that cause initialization of the PHY. Note: Port 2 driver should not try to access the PHY for configuration before this bit is set.		0h	RO

Table 28-24. Management EEPROM Control Register—EEMNGCTL [0:3] (0x1010; RO) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1010h **Offset End:** 1013h	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
19	CFG_DONE 1	Configuration cycle is done for port 1[1] – This bit indicates that configuration cycle (configuration of SerDes, PCIe* and PLLs) is done for port 1. This bit is set to 1b to indicate configuration done, and cleared by hardware on any of the reset sources that cause initialization of the PHY. Note: Port 1 driver should not try to access the PHY for configuration before this bit is set.		0h	RO
18	CFG_DONE 0	Configuration cycle is done for port 0[1] – This bit indicates that configuration cycle (configuration of SerDes, PHY, PCIe* and PLLs) is done for port 0. This bit is set to 1b to indicate configuration done, and cleared by hardware on any of the reset sources that causes initialization of the PHY. Note: Port 0 driver should not try to access the PHY for configuration before this bit is set.		0h	RO
17	EEBUSY	EEPROM Busy - This bit indicates that the EEPROM is busy processing an EEPROM transaction and shouldn't be accessed.		0h	RO
16	WRITE	Write - This bit tells the EEPROM if the current operation is read or write: 0b = read 1b = write		0h	RO
15	START	Start - Writing a 1b to this bit causes the EEPROM to start the read or write operation according to the write bit.		0h	RO
14 :00	ADDR	Address - This field is written by MNG along with Start Read or Start write to indicate the EEPROM address to read or write.		0x0	RO

1. Bit relates to physical port. If LAN Function Swap (FACTPS.LAN Function Sel = 1) is done, Software should poll CFG_DONE bit of original port to detect end of PHY configuration operation.

28.3.1.7 Management EEPROM Read/Write Data—EEMNGDATA [0:3] (0x1014; RO)

Table 28-25. Management EEPROM Read/Write Data—EEMNGDATA [0:3] (0x1014; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1014h **Offset End:** 1017h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	RDDATA	Read Data Data returned from the EEPROM read.		X	RO
15 :00	WRDATA	Write Data Data to be written to the EEPROM.		0x0	RO

28.4 Flow Control Registers

28.4.1 Detailed Register Descriptions

28.4.1.1 Flow Control Address Low—FCAL [0:3] (0x0028; RO)

Flow control packets are defined by 802.3X to be either a unique multicast address or the station address with the Ether Type field indicating PAUSE. The FCA registers provide the value hardware uses to compare incoming packets against, to determine that it should PAUSE its output.

The FCAL register contains the lower bits of the internal 48-bit Flow Control Ethernet address. All 32 bits are valid. Software need to access the High and Low registers as separate 32-bit read operations. The complete flow control multicast address is: 0x01_80_C2_00_00_01; where 0x01 is the first byte on the wire, 0x80 is the second, etc.

Note: Any packet matching the contents of {FCAH, FCAL, FCT} when CTRL.RFCE is set is acted on by the Controller. Whether flow control packets are passed to the host (software) depends on the state of the RCTL.DPF bit and whether the packet matches any of the normal filters.

Table 28-26. Flow Control Address Low—FCAL [0:3] (0x0028; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0028h **Offset End:** 002Bh	
Size: 32 bit	**Default:** 0x00C28001				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FCAL	Flow Control Address Low			0x00C28001	RO

28.4.1.2 Flow Control Address High—FCAH [0:3] (0x002C; RO)

This register contains the upper bits of the 48-bit Flow Control Ethernet address. Only the lower 16 bits of this register have meaning. The complete Flow Control address is {FCAH, FCAL}.

The complete flow control multicast address is: 0x01_80_C2_00_00_01; where 0x01 is the first byte on the wire, 0x80 is the second, etc.

Table 28-27. Flow Control Address High—FCAH [0:3] (0x002C; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 002Ch **Offset End:** 002Fh	
Size: 32 bit	**Default:** 0x00000100			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Write 0 ignore on read.			
15 :00	FCAH	Flow Control Address High Should be programmed with 0x01_00.		0100h	RO

28.4.1.3 Flow Control Type—FCT [0:3] (0x0030; R/W)

This register contains the type field that hardware matches to recognize a flow control packet. Only the lower 16 bits of this register have meaning. This register should be programmed with 0x88_08. The upper byte is first on the wire FCT[15:8].

Table 28-28. Flow Control Type—FCT [0:3] (0x0030; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0030h **Offset End:** 0033h	
Size: 32 bit	**Default:** 0x00008808			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Write 0 ignore on read.			
15 :00	FCT	Flow Control Type		0x8808	R/W

28.4.1.4 Flow Control Transmit Timer Value—FCTTV [0:3] (0x0170; R/W)

The 16-bit value in the TTV field is inserted into a transmitted frame (either XOFF frames or any PAUSE frame value in any software transmitted packets). It counts in units of slot time of 64 bytes. If software needs to send an XON frame, it must set TTV to 0b prior to initiating the PAUSE frame.

Table 28-29. Flow Control Transmit Timer Value—FCTTV [0:3] (0x0170; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0170h **Offset End:** 0173h		
Size: 32 bit	**Default:** 0x0		**Power Well:** GBEAUX		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	TTV	Transmit Timer Value		X	R/W

28.4.1.5 Flow Control Receive Threshold Low—FCRTL0 [0:3] (0x2160; R/W)

This register contains the receive threshold used to determine when to send an XON packet The complete register reflects the threshold in units of bytes. The lower 4 bits must be programmed to 0b (16 byte granularity). Software must set *XONE* to enable the transmission of XON frames. Each time hardware crosses the receive-high threshold (becoming more full), and then crosses the receive-low threshold and *XONE* is enabled (1b), hardware transmits an XON frame. When XONE is set, the *RTL* field should be programmed to at least 1b (at least 16 bytes).

Flow control reception/transmission are negotiated capabilities by the Auto-Negotiation process. When the Controller is manually configured, flow control operation is determined by the *CTRL.RFCE* and *CTRL.TFCE* bits.

Table 28-30. Flow Control Receive Threshold Low—FCRTL0 [0:3] (0x2160; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 2160h **Offset End:** 2163h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	XONE	XON Enable 0b = Disabled. 1b = Enabled.		0h	R/W
30 :17	Reserved	Reserved Write 0 ignore on read.			
16 :04	RTL	Receive Threshold Low. FIFO low water mark for flow control transmission. An XON packet is sent if the occupied space in the packet buffer is smaller or equal than this watermark. This field is in 16 bytes granularity.		0x0	R/W
03 :00	Reserved	Reserved Write 0 ignore on read.			

28.4.1.6 Flow Control Receive Threshold High—FCRTH0 [0:3] (0x2168; R/W)

This register contains the receive threshold used to determine when to send an XOFF packet. The complete register reflects the threshold in units of bytes. This value must be at maximum 48 bytes less than the maximum number of bytes allocated to the Receive Packet Buffer (*IRPBS.RXPbsize*), and the lower 4 bits must be programmed to 0b (16 byte granularity). The value of *RTH* should also be bigger than *FCRTL.RTL*. Each time the receive FIFO reaches the fullness indicated by *RTH*, hardware transmits a PAUSE frame if the transmission of flow control frames is enabled.

Flow control reception/transmission are negotiated capabilities by the Auto-Negotiation process. When the Controller is manually configured, flow control operation is determined by the *CTRL.RFCE* and *CTRL.TFCE* bits.

Table 28-31. Flow Control Receive Threshold High—FCRTH0 [0:3] (0x2168; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2168h **Offset End:** 216Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved Write 0 ignore on read.			
17 :04	RTH	Receive Threshold High FIFO high water mark for flow control transmission. An XOFF packet is sent if the occupied space in the packet buffer is bigger or equal than this watermark. This field is in 16 bytes granularity. Note: When in DMA coalescing operation threshold high value defined in *FCRTC.RTH_Coal* is used instead of RTH value to allow increase of Receive Threshold High value by maximum supported Jumbo frame size. Increase is possible since during DMA coalescing operation transmit buffer is always empty.		0x0	R/W
03 :00	Reserved	Reserved Write 0 ignore on read.			

28.4.1.7 Flow Control Refresh Threshold Value—FCRTV[0:3] (0x2460; R/W)

Table 28-32. Flow Control Refresh Threshold Value—FCRTV[0:3](0x2460; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2460h **Offset End:** 2463h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15 :00	FC_refresh_th	Flow Control Refresh Threshold This value indicates the threshold value of the flow control shadow counter; when the counter reaches this value, and the conditions for PAUSE state are still valid (buffer fullness above low threshold value), a PAUSE (XOFF) frame is sent to link partner. If this field contains zero value, the Flow Control Refresh is disabled.		0x0	R/W

28.4.1.8 Flow Control Status—FCSTS0 [0:3] (0x2464; RO)

This register describes the status of the flow control machine.

Table 28-33. Flow Control Status—FCSTS0 [0:3] (0x2464; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2464h **Offset End:** 2467h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Refresh counter	Flow control refresh counter		0x0	RO
15 :03	Reserved	Reserved			
02	Below low	The size of data in the memory is below the low threshold		X	RO
01	Above high	The size of data in the memory is above the high threshold		X	RO
00	Flow_control state	Flow control state machine signal 0b = XON 1b = XOFF		0h	RO

28.5 GbE PCIe* Registers

28.5.1 Detailed Register Description

28.5.1.1 Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO)

Firmware uses this register for configuration

Table 28-34. Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B30h **Offset End:** 5B33h	
Size: 32 bit	**Default:** 0x0			**Power Well:** CGBAhUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	PM State Changed	Indication that one or more of the functions power states had changed. This bit is also a signal to the MNG unit to create an interrupt. This bit is cleared on read, and is not set for at least 8 cycles after it was cleared.		0h	RO
30	Reserved	Reserved as 0			
29	MNGCG	MNG Clock Gated When set, indicates that the manageability clock is gated.		0h	RO
28 :18	Reserved	Reserved			
17	Func4 Aux_En	Function 4 Auxiliary (AUX) Power PM Enable bit shadow from the configuration space.		0h	RO

Table 28-34. Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B30h **Offset End:** 5B33h	
Size: 32 bit	**Default:** 0x0			**Power Well:** CGBAhUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
16	LAN3 Valid	LAN 3 Enable When set to 0b, it indicates that the LAN 3 function is disabled. When the function is enabled, the bit is set to 1b. The LAN 3 enable bit is set by the LAN3_DIS_N strapping pin. **The PCH does not support the strap pin.**		0h	RO
15 :14	Func4 Power State	Power state indication of Function 4 00b → DR 01b → D0u 10b → D0a 11b → D3		0h	RO
13	Func3 Aux_En	Function 3 Auxiliary (AUX) Power PM Enable bit shadow from the configuration space.		0h	RO
12	LAN2 Valid	LAN 2 Enable When set to 0b, it indicates that the LAN 2 function is disabled. When the function is enabled, the bit is set to 1b. The LAN 2 enable bit is set by the LAN2_DIS_N strapping pin. **The PCH does not support the strap pin.**		0h	RO
11 :10	Func3 Power State	Power state indication of Function 3 00b → DR 01b → D0u 10b → D0a 11b → D3		0h	RO
09	Func2 Aux_En	Function 2 Auxiliary (AUX) Power PM Enable bit shadow from the configuration space.		0h	RO
08	LAN1 Valid	LAN 1 Enable When set to 0b, it indicates that the LAN 1 function is disabled. When the function is enabled, the bit is set to 1b. The LAN 1 enable bit is set by the LAN1_DIS_N strapping pin. **The PCH does not support the strap pin.**		0h	RO
07 :06	Func2 Power State	Power state indication of Function 2 00b → DR 01b → D0u 10b → D0a 11b → D3		0h	RO
05 :04	Reserved	Reserved.			
03	Func1_Aux_En	Function 1Auxiliary (AUX) Power PM Enable bit shadow from the configuration space.		0h	RW

Table 28-34. Function Active and Power State to MNG—FACTPS[0:3] (0x5B30; RO) (Sheet 3 of 3)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 5B30h Offset End: 5B33h	
Size: 32 bit	Default: 0x0				Power Well: CGBAhUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
02	LAN0 Valid	LAN 0 Enable When set to 0b, it indicates that the LAN 0 function is disabled. When the function is enabled, the bit is set to 1b. The LAN 0 enable bit is set by the LAN0_DIS_N strapping pin. **The PCH does not support the strap pin.**			0h	RO
01 :00	Func1 Power State	Power state indication of Function 1 00b → DR 01b → D0u 10b →D0a 11b → D3			0h	RO

28.5.1.2 Mirrored Revision ID—MREVID[0:3] (0x5B64; R/W)

Table 28-35. Mirrored Revision ID—MREVID[0:3] (0x5B64; R/W)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 5B64h Offset End: 5B67h	
Size: 32 bit	Default: 0x0				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved				
15 :08	Step REV ID	Revision ID .			0x0	R/W
07 :00	EEPROM RevID	Mirroring of Revision ID loaded from the EEPROM in PCIe* configuration space (from word *Device Rev ID* word, address 0x1E).			0x0	R/W

28.5.1.3 PCIe* Control Extended Register—GCR_EXT[0:3] (0x5B6C; R/W)

Table 28-36. PCIe* Control Extended Register—GCR_EXT[0:3] (0x5B6C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B6Ch **Offset End:** 5B6Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved		0x0	RW
30	Reserved	Reserved		0x0	RO
29 :05	Reserved	Reserved		0x0	RW
04	APBACD	Auto PBA Clear Disable. When set to 1, Software can clear the PBA only by direct write to clear access to the PBA bit. When set to 0, any active PBA entry is cleared on the falling edge of the appropriate interrupt request to the PCIe* block. The appropriate interrupt request is cleared when software sets the associated interrupt mask bit in the EIMS (re-enabling the interrupt) or by direct write to clear to the PBA.		0x0	RW
03 :00	Reserved	Reserved			

28.6 Semaphore Registers

This section contains registers common to all ports used to coordinate between all functions. The usage of these registers is described in Section 23.5.

28.6.1 Detailed Register Descriptions

28.6.1.1 Software Semaphore—SWSM[0:3] (0x5B50; R/W)

Table 28-37. Software Semaphore—SWSM[0:3] (0x5B50; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B50h **Offset End:** 5B53h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Reserved	Reserved			
01	SWESMBI	Software/Firmware Semaphore bit This bit should be set only by the device driver (read only to firmware). The bit is not set if bit 0 in the FWSM register is set. The device driver should set this bit and than read it to verify that it was set. If it was set, it means that the device driver can access the SW_FW_SYNC register. The device driver should clear this bit after modifying the SW_FW_SYNC register. Notes: • If software takes ownership of the *SWSM.SWESMBI* bit for a duration longer than 100 mS, Firmware may take ownership of the bit. Hardware clears this bit on PCIe* reset.		0x0	R/W
00	SMBI (RS)	Software/Software Semaphore Bit This bit is set by hardware when this register is read by the device driver and cleared when the HOST driver writes a 0b to it. The first time this register is read, the value is 0b. In the next read the value is 1b (hardware mechanism). The value remains 1b until the software device driver clears it. This bit can be used as a semaphore between all the device's drivers in the Controller. This bit is cleared on PCIe* reset.		0x0	R/W

28.6.1.2 Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS)

Table 28-38. Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B54h **Offset End:** 5B57h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**[1]	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved.			
30	SERDES3_Config_ Err_Ind	SerDes3 configuration error indication Set to 1b by firmware when it fails to configure LAN3 SerDes. Cleared by firmware upon successful configuration of LAN3 SerDes.		0h	R/WS
29	SERDES2_Config_ Err_Ind	SerDes2 configuration error indication Set to 1b by firmware when it fails to configure LAN2 SerDes. Cleared by firmware upon successful configuration of LAN2 SerDes.		0h	R/WS
28	Reserved	Reserved.			
27	PHY_SERDES1_Co nfig_ Err_Ind	PHY/SerDes1 configuration error indication Set to 1b by firmware when it fails to configure LAN1 PHY/SerDes. Cleared by firmware upon successful configuration of LAN1 PHY/SerDes.		0h	R/WS
26	PHY_SERDES0_Co nfig_ Err_Ind	PHY/SerDes0 configuration error indication Set to 1b by firmware when it fails to configure LAN0 PHY/SerDes. Cleared by firmware upon successful configuration of LAN0 PHY/SerDes.		0h	R/WS
25	Reserved	Reserved			
24 :19	Ext_Err_Ind	External error indication Firmware writes here the reason that the firmware operation has stopped. For example, EEPROM CRC error, etc. Possible values: 0x00: No Error 0x01 to 0x04: Reserved. 0x05: EEPROM CRC error in SB section. 0x06: EEPROM CRC error in PT-LAN/CSRs sections. 0x07: EEPROM CRC error in FW Code section. 0x08: CRC error in PHY section. 0x09: Reserved. 0x0A: No Manageability (No EEPROM) 0x0F: Management memory Parity error. 0x10 to 0x03F: Reserved Note: When management error is detected, *ICR.MGMT* is set and an interrupts is sent to the Host. Ext_Err_ind values of 0x00 or 0x0A do not cause interrupt generation.		0x0	R/WS
18 :16	Reset_Cnt	Reset Counter Firmware increments the count on every Firmware reset. After 7 Firmware reset events counter stays stuck at 7 and does not wrap around.		0h	R/WS

Table 28-38. Firmware Semaphore—FWSM[0:3] (0x5B54; R/WS) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B54h **Offset End:** 5B57h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**[1]	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	FW_Val_Bit	Firmware Valid Bit Hardware clears this bit in reset de-assertion so software can know firmware mode (bits 1-3) bits are invalid. Firmware should set this bit to 1b when it is ready (end of boot sequence).		0h	R/WS
14 :07	Reserved	Reserved			
06	EEP_Reload_Ind	EEPROM reloaded indication Set to 1b after firmware reloads the EEPROM. Cleared by firmware once the "Clear Bit" host command is received from host software.		0h	R/WS
05 :04	Reserved	Reserved			
03 :01	FW_Mode	Firmware Mode Indicates the firmware mode as follows: 000b = No MNG. 001b = Reserved. 010b = PT mode. 011b = Reserved. 100b = Host Interface enable only.		0x0	R/WS
00	EEP_FW_Semaphore	Software/Firmware Semaphore Firmware should set this bit to 1b before accessing the *SW_FW_SYNC* register. If the software is using the SWSM register and does not lock the SW_FW_SYNC, firmware is able to set this bit to 1b. Firmware should set this bit back to 0b after modifying the *SW_FW_SYNC* register. Note: If software takes ownership of the *SWSM.SWESMBI* bit for a duration longer than 100 mS, Firmware may take ownership of the bit.		0h	R/WS

Notes:
1. This register should be written only by the manageability firmware. The device driver should only read this register.
2. Firmware ignores the EEPROM semaphore in operating system hung states.
3. Bits 15:00 are cleared on firmware reset.

28.6.1.3 Software-Firmware Synchronization—SW_FW_SYNC[0:3] (0x5B5C; RWS) Check with LAD for Phy Related Bits

This register is intended to synchronize between software and firmware. This register is common to all ports.

Note: If software takes ownership of bits in the *SW_FW_SYNC* register for a duration longer than 1 second, then firmware may take ownership of the bit.

Table 28-39. Firmware Synchronization—SW_FW_SYNC[0:3] (0x5B5C; RWS)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B5Ch **Offset End:** 5B5Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved	Reserved			
22 :20	Reserved	Reserved			
19	FW_MAC_CSR_SM	When set to 1b, firmware owns access to shared CSRs		0h	RWS
18 :17	Reserved	Reserved			
16	FW_EEP_SM	When set to 1b, EEPROM access is owned by firmware		0h	RWS
15 :09	Reserved	Reserved			
08	SW_MB_SM	When Set to 1b, *SWMBWR* mailbox write register, is owned by software driver.		0h	RWS
07	Reserved	Reserved. Write 0, ignore on read.			
06	SW_PHY_SM3	When set to 1b, SerDes/PHY 3 access is owned by software		0h	RWS
05	SW_PHY_SM2	When set to 1b, SerDes/PHY 2 access is owned by software		0h	RWS
04	Reserved	Reserved			
03	SW_MAC_CSR_SM	When set to 1b, software owns access to shared CSRs		0h	RWS
02	SW_PHY_SM1	When set to 1b, SerDes/PHY 1 access is owned by software		0h	RWS
01	SW_PHY_SM0	When set to 1b, SerDes/PHY 0 access is owned by software		0h	RWS
00	SW_EEP_SM	When set to 1b, EEPROM access is owned by software		0h	RWS

Reset conditions:

- The software-controlled bits 15:0 are reset as any other CSR on global resets, D3hot exit and Forced TCO. Software is expected to clear the bits on entry to D3 state.
- The firmware controlled bits (bits 31:16) are reset on LAN_PWR_GOOD (power-up) and firmware reset.

28.6.1.4 Software Mailbox Write—SWMBWR[0:3] (0x5B04; R/W)

Table 28-40. Software Mailbox Write—SWMBWR[0:3] (0x5B04; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B04h **Offset End:** 5B07h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Mailbox	Message sent from driver to the other drivers. The interpretation of this field is defined by the drivers. This register is reset only by power on reset.		0x0	R/W

28.6.1.5 Software Mailbox 0—SWMB0[0:3] (0x5B08; RO)

Table 28-41. Software Mailbox 0—SWMB0[0:3] (0x5B08; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B08h **Offset End:** 5B0Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Mailbox	Message sent from the driver of port 0. The interpretation of this field is defined by the drivers. This register is reset only by power on reset.		0x0	RO

28.6.1.6 Software Mailbox 1—SWMB1[0:3] (0x5B0C; RO)

Table 28-42. Software Mailbox 1—SWMB1[0:3] (0x5B0C; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B0Ch **Offset End:** 5B0Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Mailbox	Message sent from the driver of port 1. The interpretation of this field is defined by the drivers. This register is reset only by power on reset.		0x0	RO

28.6.1.7 Software Mailbox 2—SWMB2[0:3] (0x5B18; RO)

Table 28-43. Software Mailbox 2—SWMB2[0:3] (0x5B18; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B18h **Offset End:** 5B1Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Mailbox	Message sent from the driver of port 2. The interpretation of this field is defined by the drivers. This register is reset only by power on reset.		0x0	RO

28.6.1.8 Software Mailbox 3—SWMB3[0:3] (0x5B1C; RO)

Table 28-44. Software Mailbox 3—SWMB3[0:3] (0x5B1C; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5B1Ch **Offset End:** 5B1Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Mailbox	Message sent from the driver of port 3.The interpretation of this field is defined by the drivers. This register is reset only by power on reset.		0x0	RO

28.7 Interrupt Register Descriptions

28.7.1 Detailed Register Descriptions

28.7.1.1 Extended Interrupt Cause—EICR[0:3] (0x1580; RC/W1C)

This register contains the frequent interrupt conditions for the Controller. Each time an interrupt causing event occurs, the corresponding interrupt bit is set in this register. An interrupt is generated each time one of the bits in this register is set and the corresponding interrupt is enabled via the Interrupt Mask Set/Read register. The interrupt might be delayed by the selected Interrupt Throttling register.

The software device driver cannot determine from the RxTxQ bits what was the cause of the interrupt. The possible causes for asserting these bits are:

- Receive Descriptor Write Back, Receive Descriptor Minimum Threshold hit, low latency interrupt for Rx, Transmit Descriptor Write Back.

Writing a 1b to any bit in the register clears that bit. Writing a 0b to any bit has no effect on that bit.

Register bits are cleared on register read.

Auto clear can be enabled for any or all of the bits in this register.

Note: Bits are not reset by Device Reset (*CTRL.DEV_RST*).

28.7.1.2 EICR Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Table 28-45. EICR Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1580h **Offset End:** 1583h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Other Cause	Interrupt Cause Active Activated when any bit in the ICR register is set.		0h	
30	TCP Timer	TCP Timer Expired Activated when the TCP timer reaches its terminal count.		0h	
29 :08	Reserved	Reserved			
07 :00	RxTxQ	Receive/Transmit Queue Interrupts One bit per queue or a bundle of queues, activated on receive/transmit queue events for the corresponding bit, such as: Receive Descriptor Write Back, Receive Descriptor Minimum Threshold hit Transmit Descriptor Write Back. The mapping of actual queue to the appropriate RxTxQ bit is according to the IVAR registers.		0x0	

28.7.1.3 EICR Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Table 28-46. EICR Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1580h **Offset End:** 1583h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	Indicates an interrupt cause mapped to MSI-X vectors 9:0 Note: Bits are not reset by Device Reset (CTRL.DEV_RST).		0x0	

28.7.1.4 Extended Interrupt Cause Set—EICS [0:3] (0x1520; WO)

Software uses this register to set an interrupt condition. Any bit written with a 1b sets the corresponding bit in the Extended Interrupt Cause Read register. An interrupt is then generated if one of the bits in this register is set and the corresponding interrupt is enabled via the Extended Interrupt Mask Set/Read register. Bits written with 0b are unchanged.

28.7.1.5 EICS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Table 28-47. EICS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1520h **Offset End:** 1523h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Other Causes	Set when any bit in Other Causes is set		0h	
30	TCP Timer	Sets the corresponding EICR TCP Timer interrupt condition.		0h	
29 :08	Reserved	Reserved			
07 :00	RxTxQ	Sets to corresponding EICR RxTXQ interrupt condition.		0x0	

Note: To set bit 31 of the *EICR* (*Other Causes*), the ICS and IMS registers should be used in order to enable one of the legacy causes.

28.7.1.6 EICS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Table 28-48. SEICS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1520h **Offset End:** 1523h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	Sets the corresponding EICR bit of MSI-X vectors 9:0		0x0	

28.7.1.7 Extended Interrupt Mask Set/Read—EIMS [0:3] (0x1524; RWS)

Reading of this register returns which bits have an interrupt mask set. An interrupt in EICR is enabled if its corresponding mask bit is set to 1b and disabled if its corresponding mask bit is set to 0b. A PCI interrupt is generated each time one of the bits in this register is set and the corresponding interrupt condition occurs (subject to throttling). The occurrence of an interrupt condition is reflected by having a bit set in the Extended Interrupt Cause Read register.

An interrupt might be enabled by writing a 1b to the corresponding mask bit location (as defined in the EICR register) in this register. Any bits written with a 0b are unchanged. As a result, if software needs to disable an interrupt condition that had been previously enabled, it must write to the Extended Interrupt Mask Clear register rather than writing a 0b to a bit in this register.

28.7.1.8 EIMS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Table 28-49. EIMS Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1524h **Offset End:** 1527h	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Other Cause	Set Mask bit for the corresponding EICR other cause interrupt condition.		1h	
30	TCP Timer	Set Mask bit for the corresponding EICR TCP timer interrupt condition.		0h	
29 :08	Reserved	Reserved			
07 :00	RxTxQ	Set Mask bit for the corresponding EICR RxTXQ interrupt.		0x0	

Note: Bits are not reset by Device Reset (*CTRL.DEV_RST*).

28.7.1.9 EIMS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Table 28-50. EIMS Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1524h **Offset End:** 1527h	
Size: 32 bit	**Default:** Xh			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	Set Mask bit for the corresponding EICR bit of MSI-X vectors 9:0. Note: Bits are not reset by Device Reset (CTRL.DEV_RST).		0x0	

28.7.1.10 Extended Interrupt Mask Clear - EIMC [0:3] (0x1528; WO)

This register provides software a way to disable certain or all interrupts. Software disables a given interrupt by writing a 1b to the corresponding bit in this register.

On interrupt handling, the software device driver should set all the bits in this register related to the current interrupt request even though the interrupt was triggered by part of the causes that were allocated to this vector.

Interrupts are presented to the bus interface only when the mask bit is set to 1b and the cause bit is set to 1b. The status of the mask bit is reflected in the Extended Interrupt Mask Set/Read register and the status of the cause bit is reflected in the Interrupt Cause Read register.

Software blocks interrupts by clearing the corresponding mask bit. This is accomplished by writing a 1b to the corresponding bit location (as defined in the EICR register) of that interrupt in this register. Bits written with 0b are unchanged (their mask status does not change).

28.7.1.11 EIMC Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Table 28-51. EIMC Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1528h **Offset End:** 152Bh	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Other Cause	Clear Mask bit for the corresponding EICR other cause interrupt.		1h	WO
30	TCP Timer	Clear Mask bit for the corresponding EICR TCP timer interrupt.		0h	WO
29 :08	Reserved	Reserved			
07 :00	RxTxQ	Clear Mask bit for the corresponding EICR RxTXQ interrupt.		0x0	WO

28.7.1.12 EIMC Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Table 28-52. EIMC Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1528h **Offset End:** 152Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	clear Mask bit for the corresponding EICR bit of MSI-X vectors 9:0		0x0	RW

28.7.1.13 Extended Interrupt Auto Clear—EIAC [0:3] (0x152C; R/W)

This register is mapped like the EICS, EIMS, and EIMC registers, with each bit mapped to the corresponding MSI-X vector.

This register is relevant to MSI-X mode only, where read-to-clear can not be used, as it might erase causes tied to other vectors. If any bits are set in EIAC, the EICR register should not be read. Bits without auto clear set, need to be cleared with write-to-clear.

Note: EICR bits that have auto clear set are cleared by the internal emission of the corresponding MSI-X message even if this vector is disabled by the operating system.

The MSI-X message can be delayed by EITR moderation from the time the EICR bit is activated.

Table 28-53. Extended Interrupt Auto Clear—EIAC [0:3] (0x152C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 152Ch **Offset End:** 152Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	Auto clear bit for the corresponding EICR bit of MSI-X vectors 9:0. Notes: • Bits are not reset by Device Reset (CTRL.DEV_RST). • When *GPIE.Multiple_MSIX* = 0 (Non MSI-X mode) bits 8 and 9 are read only and should be ignored.		0x0	R/W

28.7.1.14 Extended Interrupt Auto Mask Enable—EIAM [0:3] (0x1530; R/W)

Each bit in this register enables clearing of the corresponding bit in EIMS following read- or write-to-clear to EICR or setting of the corresponding bit in EIMS following a write-to-set to EICS.

In MSI-X mode, this register controls which of the bits in EIMC to clear upon interrupt generation.

28.7.1.15 EIAM Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Table 28-54. IAM Register Bit Description—Non MSI-X Mode (GPIE.Multiple_MSIX = 0)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1530h **Offset End:** 1533h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Other Cause	Auto mask bit for the corresponding EICR other cause interrupt condition.		0h	R/W
30	TCP Timer	Auto mask bit for the corresponding EICR TCP timer interrupt condition.		0h	R/W
29 :08	Reserved	Reserved			
07 :00	RxTxQ	Auto Mask bit for the corresponding EICR RxTxQ interrupt.		0x0	RW

Note: Bits are not reset by Device Reset (*CTRL.DEV_RST*).

28.7.1.16 EIAM Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Table 28-55. EIAM Register Bit Description—MSI-X Mode (GPIE.Multiple_MSIX = 1)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1530h **Offset End:** 1533h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	MSIX	Auto Mask bit for the corresponding EICR bit of MSI-X vectors 9:0. Note: Bits are not reset by Device Reset (*CTRL.DEV_RST*).		0x0	RW

28.7.1.17 Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C)

This register contains the interrupt conditions for the Controller that are not present directly in the EICR. Each time an ICR interrupt causing event occurs, the corresponding interrupt bit is set in this register. The *EICR.Other* bit reflects the setting of interrupt causes from ICR as masked by the Interrupt Mask Set/Read register. Each time all un-masked causes in ICR are cleared, the *EICR.Other* bit is also cleared.

ICR bits are cleared on register read. Clear-on-read can be enabled/disabled through a general configuration register bit.

Auto clear is not available for the bits in this register.

To prevent unwanted LSC (Link Status Change) interrupts during initialization, software should disable this interrupt until the end of initialization.

Table 28-56. Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1500h **Offset End:** 1503h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	INTA	Interrupt Asserted: Indicates that the INT line is asserted. Can be used by driver in shared interrupt scenario to decide if the received interrupt was emitted by the Controller. This bit is not valid in MSI/MSI-X environments		0h	RC/W1C
30	DRSTA	Device Reset Asserted Indicates the CTRL.DEV_RST was asserted on another port or on this port. When device reset occurs all ports should re-initialize registers and descriptor rings. Note: Bit is not reset by Device Reset (CTRL.DEV_RST).		0h	RC/W1C
29	TCP timer	TCP timer interrupt		0h	RC/W1C
28	MDDET	Detected Malicious driver behavior Occurs when one of the queues used malformed descriptors. In virtualized systems, might indicate a malicious or buggy driver. **Note:** This bit should never rise during normal operation.		0h	RC/W1C
27	Reserved	Reserved			
26	Software WD	Software Watchdog This bit is set after a software watchdog timer times out.		0h	RC/W1C
25	SCE	Storm Control Event This bit is set when multicast or broadcast storm control mechanism is activated or de-activated.		0h	RC/W1C
24	CGB Exception	CGB Exception Occurred Set when any error as reported in CGB_ICAUSE register, happens. Only [24:31[and [4] are relevant for the PCH.		0h	RC/W1C
23	Reserved	Reserved Write 0, ignore on read.		0h	RC/W1C
22	FER	Fatal Error This bit is set when a fatal error is detected in one of the memories		0h	RC/W1C

Table 28-56. Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1500h **Offset End:** 1503h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
21	Reserved	Reserved			
20	OMED	Other Media Energy Detect When in SerDes/SGMII mode, indicates that link status has changed on the external media.		0h	RC/W1C
19	Time_Sync	Time_Sync Interrupt This interrupt cause is set if Interrupt is generated by the Time Sync Interrupt registers (*TSICR*, *TSIM* and *TSIS*).		0h	RC/W1C
18	MNG	Manageability Event Detected Indicates that a manageability event happened. When the Controller is at power down mode, the IPMI can generate a PME for the same events that would cause an interrupt when the Controller is in the D0 state. Note: Further information on manageability event can be found in FWSM register.		0h	RC/W1C
17 :13	Reserved	Reserved			
12	GPI_SDP1	General Purpose Interrupt on SDP1 If GPI interrupt detection is enabled on this pin (via CTRL_EXT), this interrupt cause is set when the SDP1 is sampled high.		0h	RC/W1C
11	GPI_SDP0	General Purpose Interrupt on SDP0 If GPI interrupt detection is enabled on this pin (via CTRL_EXT), this interrupt cause is set when the SDP0 is sampled high.		0h	RC/W1C
10	Reserved	Reserved			
09	MDAC	MDIO Access Complete. Set when a MDIO access or a SFP I2C transaction completes.		0h	RC/W1C
08	SWMB	Set when one of the drivers wrote a message using the SWMBWR mailbox register..		0h	RC/W1C
07	RXDW	Receiver Descriptor Write Back Set when the Controller writes back an Rx descriptor to memory.		0h	RC/W1C
06	Rx Miss	Missed packet interrupt is activated for each received packet that overflows the Rx packet buffer (overrun). The packet is dropped and also increments the associated MPC counter. Note: Could be caused by no available receive buffers or because PCIe* receive bandwidth is inadequate.		0h	RC/W1C
05	Reserved	Reserved			
04	RXDMT0	Receive Descriptor Minimum Threshold Reached Indicates that the minimum number of receive descriptors are available and software should load more receive descriptors.		0h	RC/W1C
03	Reserved	Reserved Write 0, ignore on read.			

Table 28-56. Interrupt Cause Read Register—ICR [0:3] (0x1500; RC/W1C) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1500h **Offset End:** 1503h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	Reserved	Reserved			
01	Reserved	Reserved Write 0, ignore on read.			
00	TXDW	Transmit Descriptor Written Back Set when the Controller writes back a Tx descriptor to memory.		0h	RC/W1C

28.7.1.18 Interrupt Cause Set Register—ICS [0:3] (0x1504; WO)

Software uses this register to set an interrupt condition. Any bit written with a 1b sets the corresponding interrupt. This results in the corresponding bit being set in the Interrupt Cause Read Register (see Section 28.7.1.17). A PCIe* interrupt is generated if one of the bits in this register is set and the corresponding interrupt is enabled through the Interrupt Mask Set/Read Register (see Section 28.7.1.19).

Bits written with 0 are unchanged.

Table 28-57. Interrupt Cause Set Register—ICS [0:3] (0x1504; WO) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1504h **Offset End:** 1507h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved.			
30	DRSTA	Sets the Device Reset Asserted Interrupt. When setting this bit a DRSTA interrupt is generated on this port only.		0h	WO
29	TCP timer	Sets the TCP timer interrupt.		0h	WO
28	MDDET	Sets the Detected Malicious driver behavior Interrupt.		0h	WO
27	Reserved	Reserved.			
26	Software WD	Sets the Software Watchdog Interrupt.		0h	WO
25	SCE	Set the Storm Control Event Interrupt		0h	WO
24	Reserved	Reserved			
23	Reserved	Reserved			
22	FER	Sets the Fatal Error Interrupt.		0h	WO
21	Reserved	Reserved			
20	OMED	Sets the Other Media Energy Detected Interrupt.		0h	WO

Table 28-57. Interrupt Cause Set Register—ICS [0:3] (0x1504; WO) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1504h **Offset End:** 1507h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
19	Time_Sync	Sets the Time_Sync interrupt.		0h	WO
18	MNG	Sets the Management Event Interrupt.		0h	WO
17 :14	Reserved	Reserved			
13	Reserved	Reserved			
12	GPI_SDP1	Sets the General Purpose Interrupt, related to SDP1 pin.		0h	WO
11	GPI_SDP0	Sets the General Purpose Interrupt, related to SDP0 pin.		0h	WO
10 :09	Reserved	Reserved			
08	SWMB	Sets the SWMB mailbox interrupt.		0h	WO
07	RXDW	Sets the Receiver Descriptor Write Back Interrupt.		0h	WO
06	Rx Miss	Sets the Rx Miss Interrupt.		0h	WO
05	Reserved	Reserved			
04	RXDMT0	Sets the Receive Descriptor Minimum Threshold Hit Interrupt.		0h	WO
03	Reserved	Reserved Write 0, ignore on read.		0h	WO
02	Reserved	Reserved			
01	Reserved	Reserved Write 0, ignore on read.		0h	WO
00	TXDW	Sets the Transmit Descriptor Written Back Interrupt.		0h	WO

28.7.1.19 Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W)

Reading this register returns bits that have an interrupt mask set. An interrupt is enabled if its corresponding mask bit is set to 1b and disabled if its corresponding mask bit is set to 0b. A PCIe* interrupt is generated each time one of the bits in this register is set and the corresponding interrupt condition occurs. The occurrence of an interrupt condition is reflected by having a bit set in the Interrupt Cause Read Register (see Section 28.7.1.17).

A particular interrupt can be enabled by writing a 1b to the corresponding mask bit in this register. Any bits written with a 0b are unchanged. As a result, if software desires to disable a particular interrupt condition that had been previously enabled, it must write to the Interrupt Mask Clear Register (see Section 28.7.1.20) rather than writing a 0b to a bit in this register.

Table 28-58. Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1508h **Offset End:** 150Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved.			
30	DRSTA	Sets/Reads the mask for Device Reset Asserted Interrupt. Note: Bit is not reset by Device Reset (CTRL.DEV_RST).		0h	R/W
29	TCP timer	Sets/Reads the mask for TCP timer interrupt.		0h	R/W
28	MDDET	Sets/Reads the mask for the Detected Malicious driver behavior Interrupt.		0h	R/W
27	Reserved	Reserved			
26	Software WD	Sets/Reads the mask for the Software Watchdog Interrupt.		0h	R/W
25	SCE	Sets/Reads the mask for the Storm Control Event Interrupt.		0h	R/W
24	Reserved	Reserved			
23	Reserved	Reserved Write 0, ignore on read.			
22	FER	Sets/Reads the mask for the Fatal Error Interrupt.		0h	R/W
21	Reserved	Reserved			
20	OMED	Sets/Reads the mask for Other Media Energy Detected Interrupt.		0h	R/W
19	Time_Sync	Sets/Reads the mask for Time_Sync Interrupt.		0h	R/W
18	MNG	Sets/Reads the mask for Management Event Interrupt.		0h	R/W
17 :13	Reserved	Reserved			
12	GPI_SDP1	Sets/Reads the mask for General Purpose Interrupt, related to SDP1 pin.		0h	R/W
11	GPI_SDP0	Sets/Reads the mask for General Purpose Interrupt, related to SDP0 pin.		0h	R/W
10 :09	Reserved	Reserved			
08	SWMB	Sets/Reads the mask for Mailbox interrupt.		0h	R/W
07	RXDW	Sets/Reads the mask for Receiver Descriptor Write Back Interrupt.		0h	R/W

Table 28-58. Interrupt Mask Set/Read Register—IMS [0:3] (0x1508; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1508h **Offset End:** 150Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	Rx Miss	Sets/Reads the mask for the Rx Miss Interrupt.		0h	R/W
05	Reserved	Reserved			
04	RXDMT0	Sets/Reads the mask for Receive Descriptor Minimum Threshold Hit Interrupt.		0h	R/W
03	Reserved	Reserved Write 0, ignore on read.			
02	Reserved	Reserved			
01	Reserved	Reserved Write 0, ignore on read.			
00	TXDW	Sets/Reads the mask for Transmit Descriptor Written Back Interrupt.		0h	R/W

28.7.1.20 Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO)

Software uses this register to disable an interrupt. Interrupts are presented to the bus interface only when the mask bit is set to 1b and the cause bit set to 1b. The status of the mask bit is reflected in the Interrupt Mask Set/Read Register (see Section 28.7.1.19), and the status of the cause bit is reflected in the Interrupt Cause Read Register (see Section 28.7.1.17). Reading this register returns the value of the IMS register.

Software blocks interrupts by clearing the corresponding mask bit. This is accomplished by writing a 1b to the corresponding bit in this register. Bits written with 0b are unchanged (their mask status does not change).

In interrupt handling, the software device driver should set all the bits in this register related to the current interrupt request, even though the interrupt was triggered by part of the causes that were allocated to this vector.

Table 28-59. Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 150Ch **Offset End:** 150Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved				
30	DRSTA	Clears the mask for Device Reset Asserted Interrupt.			0h	WO
29	TCP timer	Clears the mask for TCP timer interrupt.			0h	WO
28	MDDET	Clears the mask for the Detected Malicious driver behavior Interrupt.			0h	WO
27	Reserved	Reserved.			0h	WO
26	Software WD	Clears the mask for Software Watchdog Interrupt.			0h	WO
25	SCE	Clears the mask for the Storm Control Event Interrupt.			0h	WO
24	Reserved	Reserved				
23	Reserved	Reserved Write 0, ignore on read.				
22	FER	Clears the mask for the Fatal Error Interrupt.			0h	WO
21	Reserved	Reserved				
20	OMED	Clears the mask for the Other Media Energy Detected Interrupt.			0h	WO
19	Time_Sync	Clears the mask for the Time_Sync Interrupt.			0h	WO
18	MNG	Clears the mask for the Management Event Interrupt.			0h	WO
17 :13	Reserved	Reserved				
12	GPI_SDP1	Clears the mask for the General Purpose Interrupt, related to SDP1 pin.			0h	WO
11	GPI_SDP0	Clears the mask for the General Purpose Interrupt, related to SDP0 pin.			0h	WO
10 :09	Reserved	Reserved				
08	SWMB	Clears the mask for the Software Mailbox Interrupt.			0h	WO
07	RXDW	Clears the mask for the Receiver Descriptor Write Back Interrupt.			0h	WO

Table 28-59. Interrupt Mask Clear Register—IMC [0:3] (0x150C; WO) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 150Ch **Offset End:** 150Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
06	Rx Miss	Clears the mask for the Rx Miss Interrupt.			0h	WO
05	Reserved	Reserved				
04	RXDMT0	Clears the mask for Receive Descriptor Minimum Threshold Hit Interrupt.			0h	WO
03	Reserved	Reserved Write 0, ignore on read.				
02	Reserved	Reserved				
01	Reserved	Reserved Write 0, ignore on read.			0h	WO
00	TXDW	Clears the mask for Transmit Descriptor Written Back Interrupt.			0h	WO

28.7.1.21 Interrupt Acknowledge Auto Mask Register—IAM [0:3] (0x1510; R/W)

Table 28-60. Interrupt Acknowledge Auto Mask Register—IAM [0:3] (0x1510; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1510h **Offset End:** 1513h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :0	IAM_VALUE	An ICR read or write will have the side effect of writing the contents of this register to the *IMC* register. If *GPIE.NSICR* = 0, then the copy of this register to *IMS* will occur only if at least one bit is set in the IMS and there is a true interrupt as reflected in *ICR.INTA*. Note: Bit 30 of this register is not reset by Device Reset (CTRL.DEV_RST).		0h	R/W

28.7.1.22 Interrupt Throttle—EITR [0:3] (0x1680 + 4*n [n = 0...9]; R/W)

Each EITR is responsible for an interrupt cause (RxTxQ, TCP timer and Other Cause). The allocation of EITR-to-interrupt cause is through the IVAR registers.

Software uses this register to pace (or even out) the delivery of interrupts to the host processor. This register provides a guaranteed inter-interrupt delay between interrupts asserted by the Controller, regardless of network traffic conditions. To independently validate configuration settings, software can use the following algorithm to convert the inter-interrupt interval value to the common interrupts/sec. performance metric:

$$\text{interrupts/sec} = (8 * 10^{-6}\text{sec} \times \text{interval})^{-1}$$

A counter counts in units of 8*10-6 sec. After counting "interval "number of units, an interrupt is sent to the software. The above equation gives the number of interrupts per second. The equation below time in seconds between consecutive interrupts.

For example, if the interval is programmed to 125 (decimal), the Controller guarantees the processor does not receive an interrupt for 1 ms from the last interrupt. The maximum observable interrupt rate from the Controller should never exceed 7813 interrupts/sec.

Inversely, inter-interrupt interval value can be calculated as:

$$\text{inter-interrupt interval} = (8 * 10^{-6}\text{sec} \times \text{interrupt/sec})^{-1}$$

The optimal performance setting for this register is very system and configuration specific. An initial suggested range is 2 to 175 (2 - AF).

Note: Setting EITR to a non zero value can cause an interrupt cause Rx/Tx statistics miscount.

Table 28-61. Interrupt Throttle—EITR [0:3][0:9] (0x1680 + 4*n [n = 0…9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1680h at 4 **Offset End:** 1683h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	CNT_INGR (WO)	When set the hardware does not override the counters fields (ITR counter and LLI credit counter), so they keep their previous value. Relevant for the current write only and is always read as zero		0h	R/W
30 :21	Moderation Counter (RWS)	Down counter, exposes only the 10 most significant bits of the real 12-bit counter. Loaded with Interval value whenever the associated interrupt is signaled. Counts down to 0 and stops. The associated interrupt is signaled whenever this counter is zero and an associated (via the Interrupt Select register) EICR bit is set. If the CNT_INGR is not set this counter can be directly written by software at any time to alter the throttles performance.		0x0	R/W
20 :16	LL Counter (RWS)	Reflects the current credits for that EITR for LL interrupts. If the CNT_INGR is not set this counter can be directly written by software at any time to alter the throttles performance		0x0	R/W
15	LLI_EN	LLI moderation enable.		0h	R/W
14 :02	Interval	Minimum inter-interrupt interval. The interval is specified in 1 µs increments. A zero disables interrupt throttling logic.		0x0	R/W
01 :00	Reserved	Reserved			

Note: EITR register and interrupt mechanism is not reset by Device Reset (*CTRL.DEV_RST*). Occurrence of Device Reset interrupt causes immediate generation of all pending interrupts.

28.7.1.23 Interrupt Vector Allocation Registers—IVAR [0:3] (0x1700 + 4*n [n=0...3]; RW)

These registers have two modes of operation:

1. In MSI-X mode these registers define the allocation of the different interrupt causes as defined in Table 28-62 to one of the MSI-X vectors. Each INT_Alloc[i] (i=0...15) field is a byte indexing an entry in the MSI-X Table Structure and MSI-X PBA Structure.
2. In non MSI-X mode these registers define the allocation of the Rx and Tx queues interrupt causes to one of the RxTxQ bits in the EICR. Each INT_Alloc[i] (i=0…15) field is a byte indexing the appropriate RxTxQ bit as defined in Table 28-62.

Entries are mapped as follows:

a. Queues RX0, TX0, RX1, TX1 are mapped in IVAR[0] Register.
b. Queues RX2, TX2, RX3, TX3 are mapped in IVAR[1] Register.
c. Queues RX4, TX4, RX5, TX5 are mapped in IVAR[2] Register.
d. Queues RX6, TX6, RX7, TX7 in are mapped IVAR[3] Register.

Table 28-62. Interrupt Vector Allocation Registers—IVAR [0:3][0:3](0x1700 + 4*n [n=0...3]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1700h at 4 **Offset End:** 1703h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	INT_Alloc_val[3]	Valid bit for INT_Alloc[3]		0h	RW
30 :28	Reserved	Reserved			
27 :24	INT_Alloc[3]	Defines the MSI-X vector assigned to the interrupt cause associated with this entry, as defined in Table 28-62. Valid values are 0 to 9 for MSI-X mode and 0 to 7 in non MSI-X mode.		0x0	RW
23	INT_Alloc_val[2]	Valid bit for INT_Alloc[2]		0h	RW
22 :20	Reserved	Reserved			
19 :16	INT_Alloc[2]	Defines the MSI-X vector assigned to the interrupt cause associated with this entry, as defined in Table 28-62. Valid values are 0 to 9 for MSI-X mode and 0 to 7 in non MSI-X mode.		0x0	RW
15	INT_Alloc_val[1]	Valid bit for INT_Alloc[1]		0h	RW
14 :12	Reserved	Reserved			
11 :08	INT_Alloc[1]	Defines the MSI-X vector assigned to the interrupt cause associated with this entry, as defined in Table 28-62. Valid values are 0 to 9 for MSI-X mode and 0 to 7 in non MSI-X mode.		0x0	RW
07	INT_Alloc_val[0]	Valid bit for INT_Alloc[0]		0h	RW
06 :04	Reserved	Reserved			
03 :00	INT_Alloc[0]	Defines the MSI-X vector assigned to the interrupt cause associated with this entry, as defined in Table 28-62. Valid values are 0 to 9 for MSI-X mode and 0 to 7 in non MSI-X mode.		0x0	RW

DW	31 24	23 16	15 8	7 0
0	INT_ALLOC[3]	INT_ALLOC[2]	INT_ALLOC[1]	INT_ALLOC[0]
1			…	…
2		…		
3	INT_ALLOC[15]	INT_ALLOC[14]	INT_ALLOC[13]	INT_ALLOC[12]

Note: If invalid values are written to the INT_Alloc fields the result is unexpected.

28.7.1.24 Interrupt Vector Allocation Registers—MISC IVAR_MISC [0:3] (0x1740; RW)

This register is used only in MSI-X mode. This register defines the allocation of the *Other Cause* and *TCP Timer* interrupts to one of the MSI-X vectors.

Table 28-63. Interrupt Vector Allocation Registers—MISC IVAR_MISC[0:3] (0x1740; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1740h **Offset End:** 1743h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	INT_Alloc_val[17]	Valid bit for INT_Alloc[17]		0h	RW
14 :12	Reserved	Reserved			
11 :08	INT_Alloc[17]	Defines the MSI-X vector assigned to the “Other Cause” interrupt. Valid values are 0 to 9.		0x0	RW
07	INT_Alloc_val[16]	Valid bit for INT_Alloc[16]		0h	RW
06 :04	Reserved	Reserved			
03 :00	INT_Alloc[16]	Defines the MSI-X vector assigned to the TCP timer interrupt cause. Valid values are 0 to 9.		0x0	RW

28.7.1.25 General Purpose Interrupt Enable—GPIE [0:3] (0x1514; RW)

Table 28-64. General Purpose Interrupt Enable—GPIE [0:3] (0x1514; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1514h **Offset End:** 1517h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	PBA_support	PBA Support: When set, setting one of the extended interrupts masks via EIMS causes the PBA bit of the associated MSI-X vector to be cleared. Otherwise, the Controller behaves in a way that supports legacy INT-x interrupts. **Note:** Should be cleared when working in INT-x or MSI mode and set in MSI-X mode.		0h	RW
30	EIAME	Extended Interrupt Auto Mask enable: When set (usually in MSI-X mode); upon firing of an MSI-X message, bits set in EIAM associated with this message are cleared. Otherwise, EIAM is used only upon read or write of EICR/EICS registers.		0h	RW
29 :12	Reserved	Reserved			
11 :07	LL Interval	Low latency credits increment rate. The interval is specified in 4 µs increments. A value of 0x0 disables moderation of LLI for all interrupt vectors.		0x0	RW
06 :05	Reserved	Reserved			
04	Multiple MSIX	0 = on-MSI mode, or MSI-X with single vector, IVAR maps Rx/Tx causes, to 16 EICR bits, but MSIX[0] is asserted for all. 1 = MSIX mode, IVAR maps Rx/Tx causes, TCP Timer and "Other Cause" interrupts to 10 MSI-x vectors reflected in 10 EICR bits.		0h	RW
03 :01	Reserved	Reserved.			
00	NSICR	Non Selective Interrupt clear on read: When set, every read of ICR clears it. When this bit is cleared, an ICR read causes it to be cleared only if an actual interrupt was asserted or IMS = 0b.		0h	RW

28.8 MSI-X Table Register Descriptions

These registers are used to configure the MSI-X mechanism. The *message address* and *message upper address* registers sets the address for each of the vectors. The message register sets the data sent to the relevant address. The vector control registers are used to enable specific vectors.

The pending bit array register indicates which vectors have pending interrupts. The structure is listed in Table 28-65.

Table 28-65. MSI-X Table Structure

DWORD3 MSIXTVCTRL	**DWORD2 MSIXTMSG**	**DWORD1 MSIXTUADD**	**DWORD0 MSIXTADD**	**Entry Number**	**BAR 3 - Offset**
Vector Control	Msg Data	Msg Upper Addr	Msg Addr	Entry 0	Base (0x0000)
Vector Control	Msg Data	Msg Upper Addr	Msg Addr	Entry 1	Base + 1*16

Table 28-65. MSI-X Table Structure

Vector Control	Msg Data	Msg Upper Addr	Msg Addr	Entry 2	Base + 2*16
…	…	…	…	…	
Vector Control	Msg Data	Msg Upper Addr	Msg Addr	Entry (N-1)	Base + (N-1) *16

Note: N = 25.

Table 28-66. MSI-X PBA Structure

MSIXPBA[63:0]	QWORD Number	BAR 3 - Offset
Pending Bits 0 through 63	QWORD0	Base (0x2000)
Pending Bits 64 through 127	QWORD1	Base+1*8
…	…	…
Pending Bits ((N-1) div 64)*64 through N-1	QWORD((N-1) div 64)	BASE + ((N-1) div 64)*8

Note: N = 25. As a result, only QWORD0 is implemented.

28.8.1 Detailed Register Descriptions

28.8.1.1 MSIX Table Entry Lower Address—MSIXTADD [0:3][0:9] (BAR3: 0x0000 + 0x10*n [n=0…9]; R/W)

Table 28-67. MSIX Table Entry Lower Address—MSIXTADD [0:3][0:9] (BAR3: 0x0000 + 0x10*n [n=0…9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR3[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0000h at 0x10 **Offset End:** 0003h at 0x10	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	Message Address	System-Specific Message Lower Address For MSI-X messages, the contents of this field from an MSI-X table entry specifies the lower portion of the DWORD-aligned address for the memory write transaction.		0x0	R/W
01 :00	Message Address LSB (RO)	For proper DWORD alignment, software must always write 0b's to these two bits. Otherwise, the result is undefined.		0x0	R/W

28.8.1.2 MSIX Table Entry Upper Address—MSIXTUADD [0:3][0:9] (BAR3: 0x0004 + 0x10*n [n=0…9]; R/W)

Table 28-68. MSIX Table Entry Upper Address—MSIXTUADD [0:3][0:9] (BAR3: 0x0004 + 0x10*n [n=0…9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR3[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0004h at 0x10 **Offset End:** 0007h at 0x10	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Message Address	System-Specific Message Upper Address		0x0	R/W

28.8.1.3 MSIX Table Entry Message—MSIXTMSG [0:3][0:9] (BAR3: 0x0008 + 0x10*n [n=0…9]; R/W)

Table 28-69. MSIX Table Entry Message—MSIXTMSG [0:3][0:9] (BAR3: 0x0008 + 0x10*n [n=0…9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR3[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0008h at 0x10 **Offset End:** 000Bh at 0x10	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Message Data	System-Specific Message Data For MSI-X messages, the contents of this field from an MSI-X table entry specifies the data written during the memory write transaction. In contrast to message data used for MSI messages, the low-order message data bits in MSI-X messages are not modified by the function.		0x0	R/W

28.8.1.4 MSIX Table Entry Vector Control—MSIXTVCTRL [0:3][0:9] (BAR3: 0x000C + 0x10*n [n=0...9]; R/W)

Table 28-70. MSIX Table Entry Vector Control—MSIXTVCTRL [0:3][0:9] (BAR3: 0x000C + 0x10*n [n=0...9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR3[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 000Ch at 0x10 **Offset End:** 000Fh at 0x10	
Size: 32 bit	**Default:** 0x00000001			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved			
00	Mask	When this bit is set, the function is prohibited from sending a message using this MSI-X table entry. However, any other MSI-X table entries programmed with the same vector are still capable of sending an equivalent message unless they are also masked.		1h	R/W

28.8.1.5 MSIXPBA Bit Description—MSIXPBA [0:3] (BAR3: 0x2000; RO)

Table 28-71. MSIXPBA Bit Description—MSIXPBA [0:3] (BAR3: 0x2000; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR3[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 2000h **Offset End:** 2003h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	Pending Bits	For each pending bit that is set, the function has a pending message for the associated MSI-X Table entry. Pending bits that have no associated MSI-X table entry are reserved.		0x0	RO

28.8.1.6 MSIX PBA Clear—PBACL [0:3] (0x5B68; R/W1C)

Table 28-72. MSIX PBA Clear—PBACL [0:3] (0x5B68; R/W1C)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B68h **Offset End:** 5B6Bh	
Size: 32 bit	**Default:** 0X0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	PENBITCLR	MSI-X Pending bits Clear Writing a 1b to any bit clears the corresponding MSIXPBA bit; writing a 0b has no effect. **Note:** Bits are set for a single PCIe* clock cycle and than cleared.		0x0	R/W1C

28.9 Receive Registers

28.9.1 Detailed Register Descriptions

28.9.1.1 Receive Control Register—RCTL [0:3] (0x0100; R/W)

This register controls all Controller receiver functions.

Table 28-73. Receive Control Register—RCTL[0:3] (0x0100; R/W) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0100h **Offset End:** 0103h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved Should be written with 0b to ensure future compatibility.			
26	SECRC	Strip Ethernet CRC from incoming packet Causes the CRC to be stripped from all packets. 0b = Does not strip CRC 1b = Strips CRC. This bit controls whether the hardware strips the Ethernet CRC from the received packet. This stripping occurs prior to any checksum calculations. The stripped CRC is not transferred to host memory and is not included in the length reported in the descriptor.		0h	R/W
25 :24	Reserved	Reserved Should be written with 0b to ensure future compatibility.			
23	PMCF	Pass MAC Control Frames Filters out unrecognized pause and other control frames. 0b = Pass/forward pause frames. 1b = Filter pause frames (default). PMCF controls the DMA function of MAC control frames (other than flow control). A MAC control frame in this context must be addressed to either the MAC control frame multicast address or the station address, match the type field, and NOT match the PAUSE opcode of 0x0001. If PMCF = 1b then frames meeting this criteria are transferred to host memory.		0h	R/W
22	DPF	Discard Pause Frames with Station MAC Address Controls whether pause frames directly addressed to this station are forwarded to the host. 0b = incoming pause frames with station MAC address are forwarded to the host. 1b = incoming pause frames with station MAC address are discarded. **Note:** Pause frames with other MAC addresses (multicast address) are always discarded unless the specific address is added to the accepted MAC addresses (either multicast or unicast).		0h	R/W
21	PSP	Pad Small Receive packets. If this field is set, *RCTL.SECRC* should be set also.		0h	R/W
20	CFI	Canonical Form Indicator bit value 0b = 802.1Q packets with CFI equal to this field are accepted. 1b = 802.1Q packet is discarded.		0h	R/W

Table 28-73. Receive Control Register—RCTL[0:3] (0x0100; R/W) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0100h **Offset End:** 0103h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
19	CFIEN	Canonical Form Indicator Enable 0b = Disabled (CFI bit found in received 802.1Q packet's tag is not compared to decide packet acceptance). 1b = Enabled (CFI bit found in received 802.1Q packet's tag must match RCTL.CFI to accept 802.1Q type packet.		0h	R/W
18	VFE	VLAN Filter Enable 0b = Disabled (filter table does not decide packet acceptance). 1b = Enabled (filter table decides packet acceptance for 802.1Q packets). Three bits [20:18] control the VLAN filter table. The first determines whether the table participates in the packet acceptance criteria. The next two are used to decide whether the CFI bit found in the 802.1Q packet should be used as part of the acceptance criteria.		0h	R/W
17 :16	BSIZE	Receive Buffer Size BSIZE controls the size of the receive buffers and permits software to trade-off descriptor performance versus required storage space. Buffers that are 2048 bytes require only one descriptor per receive packet maximizing descriptor efficiency. 00b = 2048 Bytes. 01b = 1024 Bytes. 10b = 512 Bytes. 11b = 256 Bytes. **Notes:** 1. BSIZE not modified when RXEN is set to 1b. Set RXEN =0 when modifying the buffer size by changing this field. 2. BSIZE value only defines receive buffer size of queues with a SRRCTL.BSIZEPACKET value of 0.		0h	R/W
15	BAM	Broadcast Accept Mode. 0b = Ignore broadcast (unless it matches through exact or imperfect filters). 1b = Accept broadcast packets.		0h	R/W
14	Reserved	Reserved			
13 :12	MO	Multicast Offset Determines which bits of the incoming multicast address are used in looking up the bit vector. 00b = bits [47:36] of received destination multicast address. 01b = bits [46:35] of received destination multicast address. 10b = bits [45:34] of received destination multicast address. 11b = bits [43:32] of received destination multicast address.		0h	R/W
11 :10	Reserved	Reserved Set to 0b for compatibility.			
09 :08	Reserved	Reserved			

Table 28-73. Receive Control Register—RCTL[0:3] (0x0100; R/W) (Sheet 3 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0100h **Offset End:** 0103h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :06	LBM	Loopback mode. Controls the loopback mode of the Controller. 00b = Normal operation 01b = MAC loopback (test mode). 10b = Undefined. 11b = Loopback via internal SerDes (SerDes/SGMII/KX mode only). **Note:** PHY devices require programming for loopback operation using MDIO accesses.		0h	R/W
05	LPE	Long Packet Reception Enable 0b = Disabled. 1b = Enabled. LPE controls whether long packet reception is permitted. Hardware discards long packets if LPE is 0b. A long packet is one longer than 1522 bytes. If LPE is 1b, the maximum packet size that the Controller can receive is 9.5Kbytes.		0h	R/W
04	MPE	Multicast Promiscuous Enabled 0b = Disabled. 1b = Enabled.		0h	R/W
03	UPE	Unicast Promiscuous Enabled 0b = Disabled. 1b = Enabled.		0h	R/W
02	SBP	Store Bad Packets 0b = do not store. 1b = store bad packets. This bit controls the MAC receive behavior. A packet is required to pass the address (or normal) filtering before the SBP bit becomes effective. If SBP = 0b, then all packets with layer 1 or 2 errors are rejected. The appropriate statistic would be incremented. If SBP = 1b, then these packets are received (and transferred to host memory). The receive descriptor error field (RDESC.ERRORS) should have the corresponding bit(s) set to signal the software device driver that the packet is erred. In some operating systems the software device driver passes this information to the protocol stack. In either case, if a packet only has layer 3+ errors, such as IP or TCP checksum errors, and passes other filters, the packet is always received (layer 3+ errors are not used as a packet filter). **Note:** symbol errors before the SFD are ignored. Any packet must have a valid SFD (RX_DV with no RX_ER in 10/100/1000BASE-T mode) in order to be recognized by the Controller (even bad packets). Also, erred packets are not routed to the MNG even if this bit is set.		0h	R/W
01	RXEN	Receiver Enable The receiver is enabled when this bit is set to 1b. Writing this bit to 0b stops reception after receipt of any in progress packet. All subsequent packets are then immediately dropped until this bit is set to 1b.		0h	R/W
00	Reserved	Reserved Write to 0b for future compatibility.			

28.9.1.2 Split and Replication Receive Control—SRRCTL [0:3][0:7](0xC00C + 0x40*n [n=0...7]; R/W)

Table 28-74. Split and Replication Receive Control—SRRCTL [0:3][0:7] (0xC00C + 0x40*n [n=0...7]; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** C00Ch at 0x4 **Offset End:** C00Fh at 0x4	
Size: 32 bit	**Default:** 0x00000400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Drop_En	Drop Enabled If set, packets received to the queue when no descriptors are available to store them are dropped. The packet is dropped only if there are not enough free descriptors in the host descriptor ring to store the packet. If there are enough descriptors in the host, but they are not yet fetched by the Controller, then the packet is not dropped and there are no release of packets until the descriptors are fetched. Default is 0b for queue 0 and 1b for the other queues.		0h	R/W
30	Timestamp	Timestamp Received packet 0 - Do not place timestamp at beginning of receive buffer. 1- Place timestamp at beginning of receive buffer. Timestamp is placed only in buffers of received packets that meet the criteria defined in the *TSYNCRXCTL.Type* field, 2-tuple filters or *ETQF* registers. When set a 40 bit time stamp generated from the value in SYSTIMH and SYSTIML registers is placed in the receive buffer before the MAC header of the packets defined in the *TSYNCRXCTL.Type* field.		0h	R/W
29 :28	Reserved	Reserved. Write 0 ignore on read.			
27 :25	DESCTYPE	Defines the descriptor in Rx 000b = Legacy. 001b = Advanced descriptor one buffer. 010b = Advanced descriptor header splitting. 011b = Advanced descriptor header replication - replicate always. 100b = Advanced descriptor header replication large packet only (larger than header buffer size). 111b = Reserved.		0h	R/W
24 :20	RDMTS	Receive Descriptor Minimum Threshold Size A low latency interrupt (LLI) associated with this queue is asserted whenever the number of free descriptors becomes equal to RDMTS multiplied by 16.		0x0	R/W
19 :14	Reserved	Reserved. Write 0 ignore on read.			
13 :12	Reserved	Reserved Must be set to 00b.			

Table 28-74. Split and Replication Receive Control—SRRCTL [0:3][0:7] (0xC00C + 0x40*n [n=0...7]; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C00Ch at 0x4 **Offset End:** C00Fh at 0x4	
Size: 32 bit	**Default:** 0x00000400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11 :08	BSIZEHEADER	Receive Buffer Size for Header Buffer The value is in 64 bytes resolution. Valid value can be from 64 bytes to 960 bytes. Default buffer size is 256 bytes. This field must be greater than 0 if the value of DESCTYPE is greater or equal to 2. Note: When *SRRCTL.Timestamp* is set to 1 and the value of DESCTYPE is greater or equal to 2, BSIZEHEADER size should be equal or greater than 2 (128 bytes).		4h	R/W
07	Reserved	Reserved			
06 :00	BSIZEPACKET	Receive Buffer Size for Packet Buffer The value is in 1 KB resolution. Valid values can be from 1 KB to 127 KB. Default buffer size is 0 KB. If this field is equal 0x0, then RCTL.BSIZE determines the packet buffer size.		0h	R/W

Table 28-75. Split and Replication Receive Control—SRRCTL [0:3][1:7](0xC00C + 0x40*n [n=0...7]; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C04Ch at 0x40 **Offset End:** C04Fh at 0x40	
Size: 32 bit	**Default:** 0x80000400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Drop_En	Drop Enabled If set, packets received to the queue when no descriptors are available to store them are dropped. The packet is dropped only if there are not enough free descriptors in the host descriptor ring to store the packet. If there are enough descriptors in the host, but they are not yet fetched by the Controller, then the packet is not dropped and there are no release of packets until the descriptors are fetched. Default is 0b for queue 0 and 1b for the other queues.		1h	R/W
30	Timestamp	Timestamp Received packet 0 - Do not place timestamp at beginning of receive buffer. 1- Place timestamp at beginning of receive buffer. Timestamp is placed only in buffers of received packets that meet the criteria defined in the *TSYNCRXCTL.Type* field, 2-tuple filters or *ETQF* registers. When set a 40 bit time stamp generated from the value in SYSTIMH and SYSTIML registers is placed in the receive buffer before the MAC header of the packets defined in the *TSYNCRXCTL.Type* field.		0h	R/W
29 :28	Reserved	Reserved. Write 0 ignore on read.			

Table 28-75. Split and Replication Receive Control—SRRCTL [0:3][1:7](0xC00C + 0x40*n [n=0...7]; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C04Ch at 0x40 **Offset End:** C04Fh at 0x40	
Size: 32 bit	**Default:** 0x80000400			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
27 :25	DESCTYPE	Defines the descriptor in Rx 000b = Legacy. 001b = Advanced descriptor one buffer. 010b = Advanced descriptor header splitting. 011b = Advanced descriptor header replication - replicate always. 100b = Advanced descriptor header replication large packet only (larger than header buffer size). 101b = Reserved. 111b = Reserved.		0h	R/W
24 :20	RDMTS	Receive Descriptor Minimum Threshold Size A low latency interrupt (LLI) associated with this queue is asserted whenever the number of free descriptors becomes equal to RDMTS multiplied by 16.		0x0	R/W
19 :14	Reserved	Reserved. Write 0 ignore on read.			
13 :12	Reserved	Reserved Must be set to 00b.			
11 :08	BSIZEHEADER	Receive Buffer Size for Header Buffer The value is in 64 bytes resolution. Valid value can be from 64 bytes to 1024 bytes. Default buffer size is 256 bytes. This field must be greater than 0 if the value of DESCTYPE is greater or equal to 2. Note: When *SRRCTL.Timestamp* is set to 1 and the value of DESCTYPE is greater or equal to 2, BSIZEHEADER size should be equal or greater than 2 (128 bytes).		4h	R/W
07	Reserved	Reserved			
06 :00	BSIZEPACKET	Receive Buffer Size for Packet Buffer The value is in 1 KB resolution. Valid values can be from 1 KB to 127 KB. Default buffer size is 0 KB. If this field is equal 0x0, then RCTL.BSIZE determines the packet buffer size.		0h	R/W

28.9.1.3 Packet Split Receive Type—PSRTYPE [0:3][0:7] (0x5480 + 4*n [n=0...7]; R/W)

This register enables or disables each type of header that needs to be split. Each register controls the behavior of 2 queue.

- Packet Split Receive Type Register (queue 0) - PSRTYPE0 (0x5480)
- Packet Split Receive Type Register (queue 1) - PSRTYPE1 (0x5484)
- Packet Split Receive Type Register (queue 2) - PSRTYPE2 (0x5488)
- Packet Split Receive Type Register (queue 3) - PSRTYPE3 (0x548C)
- Packet Split Receive Type Register (queue 4) - PSRTYPE3 (0x5490)
- Packet Split Receive Type Register (queue 5) - PSRTYPE3 (0x5494)
- Packet Split Receive Type Register (queue 6) - PSRTYPE3 (0x5498)
- Packet Split Receive Type Register (queue 7) - PSRTYPE3 (0x549C)

Table 28-76. Packet Split Receive Type—PSRTYPE [0:3][0:7] (0x5480 + 4*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5480h at 0x4 **Offset End:** 5483h at 0x4	
Size: 32 bit	**Default:** 0x0007FFFE			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	Reserved	Reserved			
18	PSR_type18	Header includes MAC, (VLAN/SNAP) IPv6, UDP, NFS only		1h	R/W
17	PSR_type17	Header includes MAC, (VLAN/SNAP) IPv6, TCP, NFS only		1h	R/W
16	Reserved	Reserved			
15	PSR_type15	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, UDP, NFS only		1h	R/W
14	PSR_type14	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, TCP, NFS only		1h	R/W
13	Reserved	Reserved			
12	PSR_type12	Header includes MAC, (VLAN/SNAP) IPv4, UDP, NFS only		1h	R/W
11	PSR_type11	Header includes MAC, (VLAN/SNAP) IPv4, TCP, NFS only		1h	R/W
10	Reserved	Reserved			
09	PSR_type9	Header includes MAC, (VLAN/SNAP) IPv6, UDP only		1h	R/W
08	PSR_type8	Header includes MAC, (VLAN/SNAP) IPv6, TCP only		1h	R/W
07	PSR_type7	Header includes MAC, (VLAN/SNAP) IPv6 only		1h	R/W
06	PSR_type6	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, UDP only		1h	R/W
05	PSR_type5	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, TCP only		1h	R/W
04	PSR_type4	Header includes MAC, (VLAN/SNAP) IPv4, IPv6 only		1h	R/W
03	PSR_type3	Header includes MAC, (VLAN/SNAP) IPv4, UDP only		1h	R/W
02	PSR_type2	Header includes MAC, (VLAN/SNAP) IPv4, TCP only		1h	R/W
01	PSR_type1	Header includes MAC, (VLAN/SNAP) IPv4 only		1h	R/W
00	Reserved	Reserved			

28.9.1.4 Replicated Packet Split Receive Type—RPLPSRTYPE [0:3] (0x54C0; R/W)

This register enables or disables each type of header that needs to be split. This register controls the behavior of replicated packets.

Table 28-77. Replicated Packet Split Receive Type—RPLPSRTYPE [0:3] (0x54C0; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 54C0h **Offset End:** 54C3h	
Size: 32 bit	**Default:** 0x0007FFFE			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	Reserved	Reserved			
18	PSR_type18	Header includes MAC, (VLAN/SNAP) IPv6, UDP, NFS only		1h	R/W
17	PSR_type17	Header includes MAC, (VLAN/SNAP) IPv6, TCP, NFS only		1h	R/W
16	Reserved	Reserved			
15	PSR_type15	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, UDP, NFS only		1h	R/W
14	PSR_type14	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, TCP, NFS only		1h	R/W
13	Reserved	Reserved			
12	PSR_type12	Header includes MAC, (VLAN/SNAP) IPv4, UDP, NFS only		1h	R/W
11	PSR_type11	Header includes MAC, (VLAN/SNAP) IPv4, TCP, NFS only		1h	R/W
10	Reserved	Reserved			
09	PSR_type9	Header includes MAC, (VLAN/SNAP) IPv6, UDP only		1h	R/W
08	PSR_type8	Header includes MAC, (VLAN/SNAP) IPv6, TCP only		1h	R/W
07	PSR_type7	Header includes MAC, (VLAN/SNAP) IPv6 only		1h	R/W
06	PSR_type6	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, UDP only		1h	R/W
05	PSR_type5	Header includes MAC, (VLAN/SNAP) IPv4, IPv6, TCP only		1h	R/W
04	PSR_type4	Header includes MAC, (VLAN/SNAP) IPv4, IPv6 only		1h	R/W
03	PSR_type3	Header includes MAC, (VLAN/SNAP) IPv4, UDP only		1h	R/W
02	PSR_type2	Header includes MAC, (VLAN/SNAP) IPv4, TCP only		1h	R/W
01	PSR_type1	Header includes MAC, (VLAN/SNAP) IPv4 only		1h	R/W
00	Reserved	Reserved			

28.9.1.5 Receive Descriptor Base Address Low—RDBAL [0:3] [0:7] (0xC000 + 0x40*n [n=0...7]; R/W)

This register contains the lower bits of the 64-bit descriptor base address. The lower four bits are always ignored. The Receive Descriptor Base Address must point to a 128 byte-aligned block of data.

Note: In order to keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2800, 0x2900, 0x2A00 & 0x2B00 respectively.

Table 28-78. Receive Descriptor Base Address Low—RDBAL [0:3][0:7] (0xC000 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C000h at 0x40 **Offset End:** C000h at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	RDBAL	Receive Descriptor Base Address Low		X	R/W
06 :00	Lower_0	Ignored on writes. Returns 0x0 on reads.		0x0	R/W

28.9.1.6 Receive Descriptor Base Address High—RDBAH [0:3] [0:7] (0xC004 + 0x40*n [n=0...7]; R/W)

This register contains the upper 32 bits of the 64-bit descriptor base address.

Table 28-79. Receive Descriptor Base Address High—RDBAH [0:3][0:7] (0xC004 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C004h at 0x40 **Offset End:** C007h at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RDBAH	Receive Descriptor Base Address [63:32]		X	R/W

Note: In order to keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2804, 0x2904, 0x2A04 & 0x2B04 respectively.

28.9.1.7 Receive Descriptor Ring Length—RDLEN [0:3][0:7] (0xC008 + 0x40*n [n=0...7]; R/W)

This register sets the number of bytes allocated for descriptors in the circular descriptor buffer. It must be 128-byte aligned.

Table 28-80. Receive Descriptor Ring Length—RDLEN [0:3][0:7](0xC008 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** C008h at 0x40 **Offset End:** C00Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved Reads as 0b. Should be written to 0b for future compatibility.			
19 :07	LEN	Descriptor Ring Length (number of 8 descriptor sets)		0x0	R/W
06 :00	0	Ignore on writes. Bits 6:0 must be set to zero. Bits 4:0 always read as zero.		0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2808, 0x2908, 0x2A08 & 0x2B08 respectively.

28.9.1.8 Receive Descriptor Head—RDH [0:3][0:7] (0xC010 + 0x40*n [n=0...7]; RO)

The value in this register might point to descriptors that are still not in host memory. As a result, the host cannot rely on this value in order to determine which descriptor to process.

Table 28-81. Receive Descriptor Head—RDH [0:3][0:7] (0xC010 + 0x40*n [n=0...7]; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** C010h at 0x40 **Offset End:** C013h at 0x40	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Should be written to 0b.				
15 :00	RDH	Receive Descriptor Head			0x0	RO

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2810, 0x2910, 0x2A10 & 0x2B10 respectively.

28.9.1.9 Receive Descriptor Tail—RDT [0:3][0:7] (0xC018 + 0x40*n [n=0...7]; R/W)

This register contains the tail pointers for the receive descriptor buffer. The register points to a 16-byte datum. Software writes the tail register to add receive descriptors to the hardware free list for the ring.

Note: Writing the RDT register while the corresponding queue is disabled is ignored by the PCH.

- To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2818, 0x2918, 0x2A18& 0x2B18 respectively.

Table 28-82. Receive Descriptor Tail—RDT [0:3][0:7] (0xC018 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C018h at 0x40 **Offset End:** C01Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved. Ignore on read, write 0 for future compatibility.			
15 :00	RDT	Receive Descriptor Tail		0x0	R/W

28.9.1.10 Receive Descriptor Control—RXDCTL [0:3][0:7] (0xC028 + 0x40*n [n=0...7]; R/W)

This register controls the fetching and write-back of receive descriptors. The three threshold values are used to determine when descriptors are read from and written to host memory. The values are in units of descriptors (each descriptor is 16 bytes).

Table 28-83. Receive Descriptor Control—RXDCTL [0:3][0:7] (0xC028 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** C028h at 0x40 **Offset End:** C02Bh at 0x40	
Size: 32 bit	**Default:** 0x0000140C			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved			
26	SWFLUSH (WC)	Receive Software Flush Enables software to trigger receive descriptor write-back flushing, independently of other conditions. This bit is cleared by hardware after write-back flush is triggered (may take a number of cycles).		0h	R/W
25	ENABLE	Receive Queue Enable When set, the *Enable* bit enables the operation of the specific receive queue. 1b =Enables queue. 0b =Disables queue. Setting this bit initializes Head and Tail registers (*RDH*[n] and *RDT*[n]) of the specific queue. Until then, the state of the queue is kept and can be used for debug purposes. When disabling a queue, this bit is cleared only after all activity in the queue has stopped. Note: If queue is enabled and *RCTL.RXEN* is cleared, receive activity on the queue will not commence.		0h	R/W
24 :21	Reserved	Reserved			
20 :16	WTHRESH	Write-Back Threshold WTHRESH controls the write-back of processed receive descriptors. This threshold refers to the number of receive descriptors in the on-chip buffer that are ready to be written back to host memory. In the absence of external events (explicit flushes), the write-back occurs only after at least WTHRESH descriptors are available for write-back. Possible values for this field are 0 to 15. **Note:** Since the default value for write-back threshold is 1b, the descriptors are normally written back as soon as one cache line is available. WTHRESH must contain a non-zero value to take advantage of the write-back bursting capabilities of the Controller. **Note**: It's recommended not to place a value above 0xC in the WTHRESH field.		1h	R/W
15 :13	Reserved	Reserved			
12 :08	HTHRESH	Host Threshold Field defines when receive descriptor prefetch is performed. Each time enough valid descriptors, as defined in the HTHRESH field, are available in host memory a prefetch is performed. Possible values for this field are 0 to 16.		Ah	R/W
07 :05	Reserved	Reserved			
04 :00	PTHRESH	Prefetch Threshold PTHRESH is used to control when a prefetch of descriptors is considered. This threshold refers to the number of valid, unprocessed receive descriptors the Controller has in its on-chip buffer. If this number drops below PTHRESH, the algorithm considers pre-fetching descriptors from host memory. This fetch does not happen unless there are at least HTHRESH valid descriptors in host memory to fetch. **Note:** HTHRESH should be given a non zero value each time PTHRESH is used. Possible values for this field are 0 to 16.		Ch	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2828, 0x2928, 0x2A28 & 0x2B28 respectively.

28.9.1.11 Receive Queue Drop Packet Count—RQDPC [0:3][0:7] (0xC030 + 0x40*n [n=0...7]; RC/W)

Table 28-84. Receive Queue Drop Packet Count—RQDPC [0:3][0:7] (0xC030 + 0x40*n [n=0...7]; RC/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C030h at 0x40 **Offset End:** C033h at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RQDPC	Receive Queue drop packet count - counts the number of packets dropped by a queue due to lack of descriptors available. Note: Counter does not wrap around when reaching a value of 0xFFFFFFFF.		0x0	RC/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2830, 0x2930, 0x2A30 & 0x2B30 respectively.

- Packets dropped due to the queue being disabled may not be counted by this register.

28.9.1.12 Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W)

The Receive Checksum Control register controls the receive checksum off loading features of the Controller. The Controller supports the off loading of three receive checksum calculations: the Packet Checksum, the IP Header Checksum, and the TCP/UDP Checksum.

Note: This register should only be initialized (written) when the receiver is not enabled (only write this register when RCTL.RXEN = 0b).

Table 28-85. Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5000h **Offset End:** 5003h	
Size: 32 bit	**Default:** 0x00000300			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13	PCSD	Packet Checksum Disable The packet checksum and IP identification fields are mutually exclusive with the RSS hash. Only one of the two options is reported in the Rx descriptor. RXCSUM.PCSD Legacy Rx Descriptor (SRRCTL.DESCTYPE = 000b): 0b (checksum enable) - Packet checksum is reported in the Rx descriptor. 1b (checksum disable) - Not supported. RXCSUM.PCSD Extended or Header Split Rx Descriptor (SRRCTL.DESCTYPE not equal 000b): 0b (checksum enable) - checksum and IP identification are reported in the Rx descriptor. 1b (checksum disable) - RSS Hash value is reported in the Rx descriptor.		0h	R/W
12	IPPCSE	IP Payload Checksum Enable See PCSS description.		0h	R/W
11	CRCOFL	CRC32 Offload Enable Enables the SCTP CRC32 checksum off-loading feature. If *RXCSUM.CRCOFL* is set to 1b, the Controller calculates the CRC32 checksum and indicates a pass/fail indication to software via the CRC32 *Checksum Valid* bit (RDESC.L4I) in the *Extended Status* field of the receive descriptor. In non I/OAT, this bit is read only as 0b.		0h	R/W
10	Reserved	Reserved			

Table 28-85. Receive Checksum Control—RXCSUM [0:3] (0x5000; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5000h **Offset End:** 5003h	
Size: 32 bit	**Default:** 0x00000300			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	TUOFLD	TCP/UDP Checksum Off-load Enable		1h	R/W
08	IPOFLD	IP Checksum Off-load Enable RXCSUM.IPOFLD is used to enable the IP Checksum off-loading feature. If *RXCSUM.IPOFLD* is set to 1b, the Controller calculates the IP checksum and indicates a pass/fail indication to software via the IP Checksum Error bit (IPE) in the *Error* field of the receive descriptor. Similarly, if *RXCSUM.TUOFLD* is set to 1b, the Controller calculates the TCP or UDP checksum and indicates a pass/fail indication to software via the TCP/UDP Checksum Error bit (L4E). Similarly, if *RFCTL.IPv6_DIS* and *RFCTL.IP6Xsum_DIS* are cleared to 0b and *RXCSUM.TUOFLD* is set to 1b, the Controller calculates the TCP or UDP checksum for IPv6 packets. It then indicates a pass/fail condition in the TCP/UDP *Checksum Error* bit (*RDESC.L4E*). This applies to checksum off loading only. Supported frame types: Ethernet II Ethernet SNAP		1h	R/W
07 :00	PCSS	Packet Checksum Start Controls the packet checksum calculation. The packet checksum shares the same location as the RSS field and is reported in the receive descriptor when the RXCSUM.PCSD bit is cleared. If the RXCSUM.IPPCSE is set, the Packet checksum is aimed to accelerate checksum calculation of fragmented UDP packets. See section Section 26.1.11.2 for a detailed explanation. If RXCSUM.IPPCSE is cleared (the default value), the checksum calculation that is reported in the Rx Packet checksum field is the unadjusted 16-bit ones complement of the packet. The packet checksum starts from the byte indicated by RXCSUM.PCSS (0b corresponds to the first byte of the packet), after VLAN stripping if enabled by the CTRL.VME. For example, for an Ethernet II frame encapsulated as an 802.3ac VLAN packet and with RXCSUM.PCSS set to 14, the packet checksum would include the entire encapsulated frame, excluding the 14-byte Ethernet header (DA, SA, Type/Length) and the 4-byte VLAN tag. The packet checksum does not include the Ethernet CRC if the RCTL.SECRC bit is set. Software must make the required offsetting computation (to back out the bytes that should not have been included and to include the pseudo-header) prior to comparing the packet checksum against the L4 checksum stored in the packet checksum. The partial checksum in the descriptor is aimed to accelerate checksum calculation of fragmented UDP packets. **Note:** The PCSS value should not exceed a pointer to the IP header start. If exceeded, the IP header checksum or TCP/UDP checksum is not calculated correctly.		0x0	R/W

28.9.1.13 Receive Long Packet Maximum Length—RLPML [0:3] (0x5004; R/W)

Table 28-86. Receive Long Packet Maximum Length—RLPML [0:3] (0x5004; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5004h **Offset End:** 5007h	
Size: 32 bit	**Default:** 0x00002600			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved			
13 :00	RLPML	Maximum allowed long packet length. This length is the global length of the packet including all the potential headers of suffixes in the packet.		2600h	R/W

28.9.1.14 Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W)

Table 28-87. Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5008h **Offset End:** 500Bh	
Size: 32 bit	**Default:** 0x1			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved Write 0 ignore on read.			
19	SYNQFP	Defines the priority between SYNQF & tuple filter 0b = filter priority 1b = SYN filter priority.		0h	R/W
18	LEF	Forward Length Error Packet 0b = packet with length error are dropped. 1b = packets with length error are forwarded to the host.		0h	R/W
17 :16	Reserved	Reserved Must be set to 00b.		0h	R/W
15	Reserved	Reserved			
14	IPFRSP_DIS	IP Fragment Split Disable When this bit is set, the header of IP fragmented packets are not set.		0h	R/W
13	Reserved	Reserved			
12	Reserved	Reserved			
11	IPv6XSUM_DIS	IPv6 XSUM Disable Disables XSUM on IPv6 packets.		0h	R/W
10	IPv6_DIS	IPv6 Disable Disables IPv6 packet filtering. Any received IPv6 packet is parsed only as an L2 packet.		0h	R/W

Table 28-87. Receive Filter Control Register—RFCTL [0:3] (0x5008; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5008h **Offset End:** 500Bh	
Size: 32 bit	**Default:** 0x1			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09 :08	NFS_VER	NFS Version 00b = NFS version 2. 01b = NFS version 3. 10b = NFS version 4. 11b = Reserved for future use.		0h	R/W
07	NFSR_DIS	NFS Read Disable Disables filtering of NFS read reply headers.		0h	R/W
06	NFSW_DIS	NFS Write Disable Disables filtering of NFS write request headers.		0h	R/W
05 :00	Reserved	Reserved			

28.9.1.15 Multicast Table Array—MTA [0:3][0:127] (0x5200 + 4*n [n=0...127]; R/W)

There is one register per 32 bits of the Multicast Address Table for a total of 128 registers. Software must mask to the desired bit on reads and supply a 32-bit word on writes. The first bit of the address used to access the table is set according to the RX_CTRL.MO field.

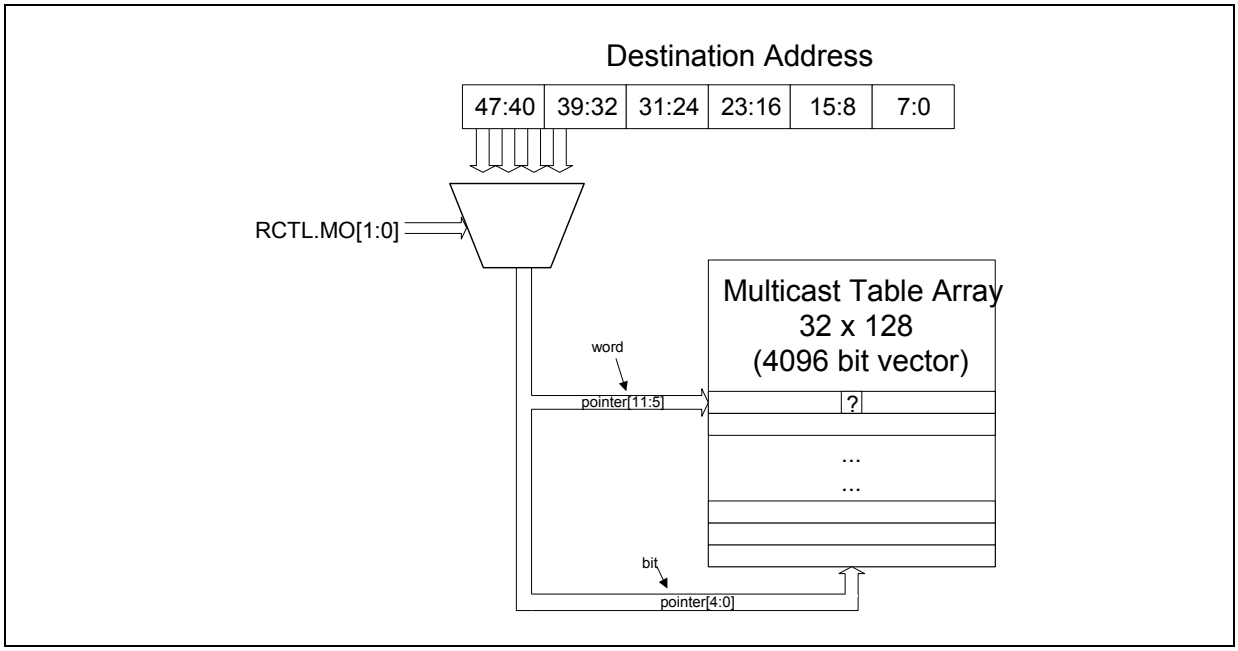
Note: All accesses to this table must be 32 bit.

Table 28-88. Multicast Table Array—MTA [0:3][0:127] (0x5200 + 4*n [n=0...127]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5200h at 0x4 **Offset End:** 5203h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Bit Vector	Word wide bit vector specifying 32 bits in the multicast address filter table.		X	R/W

Figure 28-1 shows the multicast lookup algorithm. The destination address shown represents the internally stored ordering of the received DA. Bit 0 indicated in this diagram is the first on the wire.

Figure 28-1. Multicast Table Array



28.9.1.16 Receive Address Low 0—RAL0 [0:3][0:15] (0x5400 + 8*n [n=0…15]; R/W)

While "n" is the exact unicast/multicast address entry and it is equal to 0,1,...15. These registers contain the lower bits of the 48 bit Ethernet address. All 32 bits are valid.

These registers are reset by a software reset or platform reset. The first register (RAL0) is loaded from the EEPROM after a software or platform reset.

RAL0 and RAL1 together makeup the RAL - Receive Address Low registers.

Table 28-89. Receive Address Low 0—RAL0 [0:3][0:15] (0x5400 + 8*n [n=0…15]; 0x54E0 + 8*n [n=0…7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5400h at 0x8 **Offset End:** 5403h at 0x8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RAL	Receive address low Contains the lower 32-bit of the 48-bit Ethernet address.		X	R/W

Note: The *RAL* field should be written in network order.

28.9.1.17 Receive Address Low 1—RAL1 [0:3][0:7] (0x54E0 + 8*n [n=0...7]; R/W)

While "n" is the exact unicast/multicast address entry and it is equal to 0,1,...7.

These registers contain the lower bits of the 48 bit Ethernet address. All 32 bits are valid.

These registers are reset by a software reset or platform reset. The first register (RAL0) is loaded from the EEPROM after a software or platform reset.

RAL0 and RAL1 together makeup the RAL - Receive Address Low registers.

Table 28-90. Receive Address Low 1—RAL1 [0:3][0:7] (0x54E0 + 8*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 54E0h at 0x8 **Offset End:** 54E3h at 0x8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RAL	Receive address low Contains the lower 32-bit of the 48-bit Ethernet address.		X	R/W

Note: The *RAL* field should be written in network order.

28.9.1.18 Receive Address High 0—RAH0 [0:3][0:15] (0x5404 + 8*n [n=0...15]; R/W)

These registers contain the upper bits of the 48 bit Ethernet address. The complete address is [RAH, RAL]. AV determines whether this address is compared against the incoming packet and is cleared by a master reset.

ASEL enables the Controller to perform special filtering on receive packets.

After reset, The first register (Receive Address Register 0) is loaded from the *IA* field in the EEPROM with its *Address Select* field set to 00b and its *Address Valid* field set to 1b.

Note: The *RAH* field should be written in network order.

- The first receive address register (RAH0) is also used for exact match pause frame checking (DA matches the first register). As a result, RAH0 should always be used to store the individual Ethernet MAC address of the Controller.

RAH0 and RAH1 together makeup the RAH - Receive Address High registers.

Table 28-91. Receive Address High 0—RAH0 [0:3][0:15] (0x5404 + 8*n [n=0...15]; 0x54E04 + 8*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5404h at 0x8 **Offset End:** 5407h at 0x8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Addr_Valid	Address Valid Cleared after master reset. The Address Valid field of the Receive Address Register 0 is set to 1b after a software or PCI reset or EEPROM read. In entries 0-15 this bit is cleared by master reset.		0x0	R/W
30 :26	Reserved	Reserved Write 0 Ignore on reads.			
25 :18	POOLSEL	Pool Select In virtualization modes (*MRQC.Multiple Receive Queues Enable* = 011b) indicates which Pool should get the packets matching this MAC address. This field is a bit map (bit per VM) where more than one bit can be set according to the limitations defined in Section 26.8.3.4. If all the bits are zero, this address is used only for L2 filtering and is not used as part of the queueing decision.		0x0	R/W
17 :16	ASEL	Address Select Selects how the address is to be used in the address filtering. 00b = Destination address (required for normal mode) 01b = Source address. This mode should not be used in virtualization mode. 10b = Reserved 11b = Reserved		X	R/W
15 :00	RAH	Receive address High Contains the upper 16 bits of the 48-bit Ethernet address.		X	R/W

28.9.1.19 Receive Address High 1—RAH1 [0:3][0:7] (0x54E4 + 8*n [n=0...7]; R/W)

RAH0 and RAH1 together makeup the RAH - Receive Address High registers.

Table 28-92. Receive Address High 1—RAH1 [0:3][0:7] (0x54E4 + 8*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 54E4h at 0x8 **Offset End:** 54E7h at 0x8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Addr_Valid	Address Valid Cleared after master reset. If an EEPROM is present, the *Address Valid* field of the Receive Address Register 0 is set to 1b after a software or PCI reset or EEPROM read. In entries 0-15 this bit is cleared by master reset.		0x0	R/W
30 :26	Reserved	Reserved Write 0 Ignore on reads.			
25 :18	POOLSEL	Pool Select In virtualization modes (*MRQC.Multiple Receive Queues Enable* = 011b) indicates which Pool should get the packets matching this MAC address. This field is a bit map (bit per VM) where more than one bit can be set according to the limitations defined in Section 26.8.3.4. If all the bits are zero, this address is used only for L2 filtering and is not used as part of the queueing decision.		0x0	R/W
17 :16	ASEL	Address Select Selects how the address is to be used in the address filtering. 00b = Destination address (required for normal mode) 01b = Source address. This mode should not be used in virtualization mode. 10b = Reserved 11b = Reserved		X	R/W
15 :00	RAH	Receive address High Contains the upper 16 bits of the 48-bit Ethernet address.		X	R/W

28.9.1.20 VLAN Filter Table Array—VFTA [0:3][0:127] (0x5600 + 4*n [n=0…127]; R/W)

There is one register per 32 bits of the VLAN Filter Table. The size of the word array depends on the number of bits implemented in the VLAN Filter Table. Software must mask to the desired bit on reads and supply a 32-bit word on writes.

Note: All accesses to this table must be 32 bit.

The algorithm for VLAN filtering using the VFTA is identical to that used for the Multicast Table Array. See Section 28.9.1.15 for a block diagram of the algorithm. If VLANs are not used, there is no need to initialize the VFTA.

Table 28-93. VLAN Filter Table Array—VFTA [0:3][0:127] (0x5600 + 4*n [n=0…127]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5600h at 0x4 **Offset End:** 5603h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Bit Vector	Double-word wide bit vector specifying 32 bits in the VLAN Filter table.		X	R/W

28.9.1.21 Multiple Receive Queues Command Register—MRQC [0:3] (0x5818; R/W)

The *MRQC.Multiple Receive Queues Enable* field is used to enable/disable RSS hashing and also to enable multiple receive queues. Disabling this feature is not recommended. Model usage is to reset the Controller after disabling the RSS.

Table 28-94. Multiple Receive Queues Command Register—MRQC [0:3](0x5818; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5818h **Offset End:** 581Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	RSS Field Enable	Each bit, when set, enables a specific field selection to be used by the hash function. Several bits can be set at the same time. Bit[16] = Enable TcpIPv4 hash function Bit[17] = Enable IPv4 hash function Bit[18] = Enable TcpIPv6Ex hash function Bit[19] = Enable IPv6Ex hash function Bit[20] = Enable IPv6 hash function Bit[21] = Enable TCPIPv6 hash function Bit[22] = Enable UDPIPv4 Bit[23] = Enable UDPIPv6 Bit[24] = Enable UDPIPv6Ext Bit[25] = Reserved Bits[31:26] - Reserved		0x0	R/W

Table 28-94. Multiple Receive Queues Command Register—MRQC [0:3](0x5818; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5818h **Offset End:** 581Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :06	Reserved	Reserved.			
05 :03	Def_Q	Defines the default queue in non VMDq mode according to value of the *Multiple Receive Queues Enable* field. If Multiple Receive Queues Enable = 000b: Def_Q defines the destination of all packets not forwarded by filters. 001b: 010b: Def_Q defines the destination of all packets not forwarded by RSS or filters. 011b - Def_Q field is ignored. Queueing decision of all packets not forwarded by MAC address and Ether-type filters is according to *VT_CTL.DEF_PL* field. 100-101b: Def_Q field is ignored. 110b: Note: In VMDq mode (*Multiple Receive Queues Enable* = 011b) the default queue is set according to the *VT_CTL.DEF_PL* if packet passes MAC Address filtering of a filter with *RAH.POOLSEL* = 0x0 or is a broadcast or multicast packet and does not match Ether-type queuing decision filters.		0x0	R/W
02 :00	Multiple Receive Queues Enable	Multiple Receive Queues Enable Enables support for Multiple Receive Queues and defines the mechanism that controls queue allocation. 000b = Multiple receive queues as defined by filters (2-tuple filters, L2 Ether-type filters, SYN filter and Flex Filters). 001b = 010b = Multiple receive queues as defined by filters and RSS for 8 queues[1]. 011b = Multiple receive queues as defined by VMDq based on packet destination MAC address (RAH.POOLSEL) and Ether-type queuing decision filters. 100b = 101b = Reserved 110b = 111b = Reserved. If VT is not supported, the only allowed values for this field are 000b and 010b. Writing any other value is ignored. Allowed values for this field are 000b, 010b and 011b. Any other value is ignored.		0x0	R/W

1. The *RXCSUM.PCSD* bit should be set to enable reception of the RSS hash value in the receive descriptor.

28.9.1.22 RSS Random Key Register—RSSRK [0:3][0:9] (0x5C80 + 4*n [n=0...9]; R/W)

Table 28-95. RSS Random Key Register—RSSRK [0:3][0:9] (0x5C80 + 4*n [n=0...9]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5C80h at 0x4 **Offset End:** 5C83h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	K3	Byte n*4+3 of the RSS random key (n=0,1,...9).		0x0	R/W
23 :16	K2	Byte n*4+2 of the RSS random key (n=0,1,...9).		0x0	R/W
15 :08	K1	Byte n*4+1 of the RSS random key (n=0,1,...9).		0x0	R/W
07 :00	K0	Byte n*4 of the RSS random key (n=0,1,...9).		0x0	R/W

The RSS Random Key Register stores a 40 byte key used by the RSS hash function.

31 **24**	**23** **16**	**15** **8**	**7** **0**
K[3]	K[2]	K[1]	K[0]
...	...	...	...
K[39]	...	...	K[36]

28.9.1.23 Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W)

The redirection table is a 128-entry table with each entry being eight bits wide. Only 1 to 3 bits of each entry are used to store the queue index. The table is configured through the following R/W registers.

Table 28-96. Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5C00h at 0x4 **Offset End:** 5C03h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Entry 3	Determines the tag value and physical queue for index 4*n+3 (n=0...31).		0x0	R/W

Table 28-96. Redirection Table—RETA [0:3][0:31] (0x5C00 + 4*n [n=0...31]; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5C00h at 0x4 **Offset End:** 5C03h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :16	Entry 2	Determines the tag value and physical queue for index 4*n+2 (n=0...31).		0x0	R/W
15 :08	Entry 1	Determines the tag value and physical queue for index 4*n+1 (n=0...31).		0x0	R/W
07 :00	Entry 0	Determines the tag value and physical queue for index 4*n+0 (n=0...31).		0x0	R/W

31	**24**	**23**	**16**	**15**	**8**	**7**	**0**
Tag 3		Tag 2		Tag 1		Tag 0	
...		...		...		...	
Tag 127		...		...		...	

Each entry (byte) of the redirection table contains the following:

7:3	**2:0**
Reserved	Queue index

- Bits [7:3] - Reserved
- Bits [2:0] - Queue index for all pools or in regular RSS. In RSS only mode, all bits are used.

The contents of the redirection table are not defined following reset of the Memory Configuration Registers. System software must initialize the Table prior to enabling multiple receive queues. It can also update the redirection table during run time. Such updates of the table are not synchronized with the arrival time of received packets. Therefore, it is not guaranteed that a table update takes effect on a specific packet boundary.

Note: In case the operating system provides a redirection Table whose size is smaller than 128 bytes, the software usually replicates the operating system-provided redirection table to span the whole 128 bytes of the hardware's redirection table.

28.10 Filtering Register

28.10.1 Detailed Register Descriptions

28.10.1.1 Immediate Interrupt RX—IMIR [0:3][0:7] (0x5A80 + 4*n [n=0...7]; R/W)

This register defines the filtering that corrects which packet triggers low latency interrupt. The following *IMIREXT* register includes a size threshold and a control bits bitmap to trigger an immediate interrupt.

Note: The *Port* field should be written in network order.

- If one of the actions for this filter is set, then at least one of the *PORT_BP*, *Size_BP*, one of the Mask bits or *CtrlBit_BP* bits should be cleared.

Table 28-97. Immediate Interrupt RX—IMIR [0:3][0:7] (0x5A80 + 4*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5A80h at 0x4 **Offset End:** 5A83h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Filter Priority	Defines the priority of the filter assuming two filters with same priority don't match. If two filters with the same priority match the incoming packet, the first filter (lowest ordinal number) is used in order to define the queue destination of this packet.		0h	R/W
28 :18	Reserved	Reserved			
17	PORT_BP	Port Bypass When set to 1b, the TCP port check is bypassed and only other conditions are checked. When set to 0b, the TCP port is checked to fit the port field.		X	R/W
16	Immediate Interrupt	Enables issuing an immediate interrupt when the following conditions are met: • The filter associated with this register matches • The Length filter associated with this filter matches • The TCP flags filter associated with this filter matches		0h	R/W
15 :00	Destination Port	Destination TCP port This field is compared with the Destination TCP port in incoming packets.		0x0	R/W

28.10.1.2 Immediate Interrupt Rx Ext.—IMIREXT [0:3][0:7] (0x5AA0 + 4*n [n=0…7]; R/W)

Table 28-98. Immediate Interrupt Rx Ext.—IMIREXT [0:3][0:7] (0x5AA0 + 4*n [n=0…7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5AA0h at 0x4 **Offset End:** 5AA3h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved			
19	CtrlBit_BP	Control Bits Bypass When set to 1b, the control bits check is bypassed. When set to 0b, the control bits check is performed.		X	R/W
18 :13	CtrlBit	Control Bit When a bit in this field equals 1b, an interrupt is immediately issued after receiving a packet with the corresponding TCP control bits turned on. Bit 13 (URG): Urgent pointer field significant Bit 14 (ACK): Acknowledgment field Bit 15 (PSH): Push function Bit 16 (RST): Reset the connection Bit 17 (SYN): Synchronize sequence numbers Bit 18 (FIN): No more data from sender		X	R/W
12	Size_BP	Size Bypass When 1b, the size check is bypassed. When 0b, the size check is performed.		X	R/W
11 :00	Size_Thresh	Size Threshold These 12 bits define a size threshold; a packet with a length below this threshold triggers an interrupt. Enabled by Size_Thresh_en.		X	R/W

Note: The size used for this comparison is the size of the packet as forwarded to the host and does not include any of the fields stripped by the MAC (VLAN or CRC). As a result, setting the *RCTL.SECRC* & *CTRL.VME* bits should be taken into account while calculating the size threshold.

- The value of the IMIR and *IMIREXT* registers after reset is unknown (apart from the *IMIR.PORT_IM_EN* bit which is guaranteed to be cleared). Therefore, both registers should be programmed before I*MIR.PORT_IM_EN* is set for a given flow.

28.10.1.3 2tuples Queue Filter—TTQF [0:3][0:7] (0x59E0 + 4*n[n=0..7];RW)

Table 28-99. 2tuples Queue Filter—TTQF[0:3][0:7] (0x59E0 + 4*n[n=0..7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 59E0h at 0x4 **Offset End:** 59E3h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Mask	Mask bits for the 2-tuple fields . The corresponding field participates in the match if the bit below is cleared: Bit 28 - Mask protocol comparison Bits 31 - 29 Reserved		0xF	
27	1588 Time Stamp	When set, packets that match this filter are time stamped according to the IEEE 1588 specification		0x0	R/W
26 :19	Reserved	Reserved		0h	R/W
18 :16	Rx Queue	Identifies the Rx queue associated with this 2-tuple filter		0h	R/W
15	Reserved_1	Reserved Write 1 ignore on read.			
14 :12	Reserved	Reserved			
11 :09	Reserved	Reserved Write 0 ignore on read.			
08	Queue Enable	When set, enables filtering of Rx packets by the 2-tuples defined in this filter to the queue indicated in this register.		0h	R/W
07 :00	Protocol	IP L4 Protocol, part of the 2-tuple queue filters.		0h	R/W

28.10.1.4 Immediate Interrupt Rx VLAN Priority—IMIRVP [0:3] (0x5AC0; R/W)

Table 28-100.Immediate Interrupt Rx VLAN Priority—IMIRVP [0:3] (0x5AC0; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5AC0h **Offset End:** 5AC3h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved				
03	Vlan_pri_en	VLAN Priority Enable When set to 1b, an incoming packet with VLAN tag with a priority equal or higher to Vlan_Pri triggers an immediate interrupt, regardless of the EITR moderation. When set to 0b, the interrupt is moderated by EITR.			0h	R/W
02 :00	Vlan_Pri	VLAN Priority This field includes the VLAN priority threshold. When Vlan_pri_en is set to 1b, then an incoming packet with a VLAN tag with a priority field equal or higher to VlanPri triggers an immediate interrupt, regardless of the EITR moderation.			0h	R/W

28.10.1.5 SYN Packet Queue Filter—SYNQF [0:3] (0x55FC; RW)

Table 28-101.SYN Packet Queue Filter—SYNQF [0:3] (0x55FC; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 55FCh **Offset End:** 55FFh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved				
03 :01	Rx Queue	Identifies an Rx queue associated with SYN packets.			0x0	R/W
00	Queue Enable	When set, enables forwarding of Rx packets to the queue indicated in this register.			0h	R/W

28.10.1.6 EType Queue Filter—ETQF [0:3][0:7] (0x5CB0 + 4*n[n=0...7]; R/W)

Table 28-102.EType Queue Filter—ETQF [0:3][0:7] (0x5CB0 + 4*n[n=0...7]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5CB0h at 0x4 **Offset End:** 5CB3h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Queue Enable	When set, enables filtering of Rx packets by the EType defined in this register to the queue indicated in this register.		0h	RW
30	1588 time stamp	When set, packets with this EType are time stamped according to the IEEE 1588 specification.		0h	RW
29	Immediate Interrupt	When set, packets that match this filter generate an immediate interrupt.		0x0	RW
28	Reserved	Reserved			
27	Reserved	Reserved			
26	Filter enable	When set, this filter is valid. Any of the actions controlled by the following fields are gated by this field.		0h	RW
25 :19	Reserved	Reserved			
18 :16	Rx Queue	Identifies the receive queue associated with this EType.		0x0	RW
15 :00	EType	Identifies the protocol running on top of IEEE 802. Used to forward Rx packets containing this EType to a specific Rx queue.		0x0	RW

28.11 Transmit Registers

28.11.1 Detailed Register Descriptions

28.11.1.1 Transmit Control Register—TCTL [0:3] (0x0400; R/W)

This register controls all transmit functions for the Controller.

Table 28-103.Transmit Control Register—TCTL [0:3] (0x0400; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0400h **Offset End:** 0403h	
Size: 32 bit	**Default:** 0x000400F8			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved			
24	RTLC	Re-transmit on Late Collision When set, enables the Controller to re-transmit on a late collision event. Note: RTLC configures the Controller to perform retransmission of packets when a late collision is detected. The collision window is speed dependent: 64 bytes for 10/100 Mb/s and 512 bytes for 1000 Mb/s operation. If a late collision is detected when this bit is disabled, the transmit function assumes the packet has successfully transmitted. This bit is ignored in full-duplex mode.		0h	R/W
23	Reserved	Reserved			
22	SWXOFF	Software XOFF Transmission When set to 1b, the Controller schedules the transmission of an XOFF (PAUSE) frame using the current value of the PAUSE timer (FCTTV.TTV). This bit self-clears upon transmission of the XOFF frame. Note: While 802.3x flow control is only defined during full duplex operation, the sending of PAUSE frames via the SWXOFF bit is not gated by the duplex settings within the Controller. Software should not write a 1b to this bit while the Controller is configured for half-duplex operation.		0h	R/W
21 :12	BST	Back-Off Slot Time This value determines the back-off slot time value in byte time.		40h	R/W
11 :04	CT	Collision Threshold This determines the number of attempts at retransmission prior to giving up on the packet (not including the first transmission attempt). While this can be varied, it should be set to a value of 15 in order to comply with the IEEE specification requiring a total of 16 attempts. The Ethernet back-off algorithm is implemented and clamps to the maximum number of slot-times after 10 retries. This field only has meaning when in half-duplex operation. Note: Software can choose to abort packet transmission in less than the Ethernet mandated 16 collisions. For this reason, hardware provides CT support.		Fh	R/W
03	PSP	Pad Short Packets 0b = Do not pad. 1b = Pad. Padding makes the packet 64 bytes long. This is not the same as the minimum collision distance. If padding of short packets is allowed, the total length of a packet not including FCS should be not less than 17 bytes.		1h	R/W

Table 28-103.Transmit Control Register—TCTL [0:3] (0x0400; R/W) (Sheet 2 of 2)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 0400h Offset End: 0403h	
Size: 32 bit	Default: 0x000400F8				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
02	Reserved	Reserved				
01	EN	Transmit Enable The transmitter is enabled when this bit is set to 1b. Writing 0b to this bit stops transmission after any in progress packets are sent. Data remains in the transmit FIFO until the device is re-enabled. Software should combine this operation with reset if the packets in the TX FIFO should be flushed.			0h	R/W
00	Reserved	Reserved Write as 0b for future compatibility.				

28.11.1.2 Transmit Control Extended—TCTL_EXT [0:3] (0x0404; R/W)

This register controls late collision detection.

The COLD field is used to determine the latest time in which a collision indication is considered as a valid collision and not a late collision. When using an SGMII connected external PHY, the SGMII interface adds some delay on top of the time budget allowed by the specification (collisions in valid network topographies even after 512-bit time can be expected). In order to accommodate this condition, COLD should be updated to take the SGMII inbound and outbound delays.

Table 28-104.Transmit Control Extended—TCTL_EXT [0:3] (0x0404; R/W)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 0404h Offset End: 0407h	
Size: 32 bit	Default: 0x00010840				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved.				
19 :10	COLD	Collision Distance Used to determine the latest time in which a collision indication is considered as a valid collision and not a late collision.			0x42	R/W
09 :00	Reserved	Reserved				

28.11.1.3 Transmit IPG Register—TIPG (0x0410; R/W)

This register controls the Inter Packet Gap (IPG) timer.

Table 28-105.Transmit IPG Register—TIPG [0:3] (0x0410; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0410h **Offset End:** 0413h	
Size: 32 bit	**Default:** 0x00601008			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :30	Reserved	Reserved Read as 0b. Should be written with 0b for future compatibility.			
29 :20	IPGR	IPG After Deferral Specifies the total IPG time for non back-to-back transmissions (transmission following deferral) in half duplex. Measured in increments of the MAC clock: 8 ns MAC clock when operating @ 1 Gb/s. 80 ns MAC clock when operating @ 100 Mb/s 800 ns MAC clock when operating @ 10 Mb/s. An offset of 5-byte times must be added to the programmed value to determine the total IPG after a defer event. A value of 7 is recommended to achieve a 12-byte effective IPG. The IPGR must never be set to a value greater than IPGT. If IPGR is set to a value equal to or larger that IPGT, it overrides the IPGT IPG setting in half duplex resulting in inter-packet gaps that are larger then intended by IPGT. In this case, full duplex is unaffected and always relies on IPGT.		0x06	R/W
19 :10	IPGR1	IPG Part 1 Specifies the portion of the IPG in which the transmitter defers to receive events. IPGR1 should be set to 2/3 of the total effective IPG (8). Measured in increments of the MAC clock: 8 ns MAC clock when operating @ 1 Gb/s. 80 ns MAC clock when operating @ 100 Mb/s 800 ns MAC clock when operating @ 10 Mb/s.		0x04	R/W
09 :00	IPGT	IPG Back to Back Specifies the IPG length for back to back transmissions in both full and half duplex. Measured in increments of the MAC clock: 8 ns MAC clock when operating @ 1 Gb/s. 80 ns MAC clock when operating @ 100 Mb/s. 800 ns MAC clock when operating @ 10 Mb/s. IPGT specifies the IPG length for back-to-back transmissions in both full duplex and half duplex. An offset of 4 byte times is added to the programmed value to determine the total IPG. As a result, a value of 8 is recommended to achieve a 12 byte time IPG.		0x08	R/W

28.11.1.4 Retry Buffer Control—RETX_CTL [0:3] (0x041C; R/W)

This register controls the collision retry buffer.

Table 28-106.Retry Buffer Control—RETX_CTL [0:3] (0x041C; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 041Ch **Offset End:** 041Fh	
Size: 32 bit	**Default:** 0x00000003				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved				
03 :00	Water Mark	Retry buffer water mark. This parameters defines the minimal number of QWords that should be present in the retry buffer before transmission is started.			0x3	RW

28.11.1.5 DMA Tx Control—DTXCTL [0:3] (0x3590; R/W)

This register is used to set some parameters controlling the DMA Tx behavior.

Table 28-107.DMA Tx Control—DTXCTL [0:3] (0x3590; R/W) (Sheet 1 of 2)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3590h **Offset End:** 3593h	
Size: 32 bit	**Default:** 0x00400004				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :26	Reserved	Reserved				
25 :20	Cswthresh	Context switch threshold. Defines the amount of back to back transmit context descriptors above which the driver will be considered as malicious. Maximum value should be less than 19.			0x4	R/W
19 :06	Reserved	Reserved				
05	MDP_EN	Malicious driver protection enable 0b = mechanism is disabled. 1b = mechanism is enabled.			0h	R/W
04	OutOfSyn-cEnable	0b = Out Of sync mechanism is disabled. 1b = Out Of sync mechanism is enabled.			0h	R/W

Table 28-107.DMA Tx Control—DTXCTL [0:3] (0x3590; R/W) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3590h **Offset End:** 3593h	
Size: 32 bit	**Default:** 0x00400004				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
03	Reserved	Reserved				
02	8023LL	802.3 length location 1b = The location of the 802.3 length field in 802.3+SNAP packets, is assumed to be 8 bytes before the end of the MAC header. 0b = The location of the 802.3 length field in 802.3+SNAP packets, is calculated from the beginning of the MAC header assuming no VLAN present in the packet sent by the software. This bit is used only in case of large send (TSO) with SNAP mode.			1h	R/W
01 :00	Reserved	Reserved.				

28.11.1.6 DMA TX TCP Flags Control Low—DTXTCPFLGL [0:3] (0x359C; R/W)

This register holds the buses that "AND" the control flags in TCP header for the first and middle segments of a TSO packet.

Table 28-108.DMA TX TCP Flags Control Low—DTXTCPFLGL [0:3] (0x359C; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 359Ch **Offset End:** 359Fh	
Size: 32 bit	**Default:** 0x00400004				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	Reserved	Reserved				
27 :16	TCP_Flg_mid_seg	TCP Flags middle segments. Bits that are used to execute an AND operation with the TCP flags in the TCP header in the middle segments			0xF76	RW
15 :12	Reserved	Reserved				
11 :00	TCP_flg_first_seg	TCP Flags first segment. Bits that are used to execute an AND operation with the TCP flags in the TCP header in the first segment			0xFF6	RW

28.11.1.7 DMA TX TCP Flags Control High—DTXTCPFLGH [0:3] (0x35A0; RW)

This register holds the buses that "AND" the control flags in TCP header for the last segment of a TSO packet. See Section 26.2.4.7.3 for details of use of this register.

Table 28-109.DMA TX TCP Flags Control High—DTXTCPFLGH [0:3] (0x35A0; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 35A0h **Offset End:** 35A3h	
Size: 32 bit	**Default:** 0x00000F7F			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved.			
11 :00	TCP_Flg_lst_seg	TCP Flags last segment. Bits that are used to execute an AND operation with the TCP flags at TCP header in the last segment		0xF7F	RW

28.11.1.8 DMA TX Max Total Allow Size Requests—DTXMXSZRQ [0:3] (0x3540; RW)

This register limits the total number of data bytes that might be in outstanding PCIe* requests from the host memory. This allows requests to send low latency packets to be serviced in a timely manner, as this request is serviced right after the current outstanding requests are completed.

Table 28-110.DMA TX Max Total Allow Size Requests—DTXMXSZRQ [0:3] (0x3540; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3540h **Offset End:** 3543h	
Size: 32 bit	**Default:** 0x00000010			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11 :00	Max_bytes_num_req	Maximum allowable size of concurrent TX outstanding requests on PCIe*. Field defines maximum size in 256 byte resolution of outstanding TX requests to be sent on PCIe*. If total amount of outstanding TX requests is higher than defined in this field, no further TX outstanding requests are sent.		0x10	rW

28.11.1.9 DMA TX Maximum Packet Size—DTXMXPKTSZ [0:3] (0x355C; RW)

This register limits the total number of data bytes that might be transmitted in a single frame. Reducing packet size enables better utilization of transmit buffer.

Table 28-111.DMA TX Maximum Packet Size—DTXMXPKTSZ [0:3] (0x355C; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 355Ch **Offset End:** 355Fh	
Size: 32 bit	**Default:** 0x00000098				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved				
08 :00	MAX_TPKT_SIZE	Maximum transmit packet size that is allowed to be transmitted by the driver. Value entered is in 64 Bytes resolution. Default value enables transmission of maximum sized 9728 Byte Jumbo frames.			0x98	RW

28.11.1.10 Transmit Descriptor Base Address Low—TDBAL [0:3][0:7] (0xE000 + 0x40*n [n=0...7]; R/W)

These registers contain the lower 32 bits of the 64-bit descriptor base address. The lower 7 bits are ignored. The Transmit Descriptor Base Address must point to a 128-byte aligned block of data.

Table 28-112.Transmit Descriptor Base Address Low—TDBAL [0:3][0:7] (0xE000 + 0x40*n [n=0...7]; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** E000h **Offset End:** E003h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	TDBAL	Transmit Descriptor Base Address Low			X	R/W
06 :00	Lower_0	Ignored on writes. Returns 0x0 on reads.			0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3800, 0x3900, 0x3A00 & 0x3B00 respectively.

28.11.1.11 Transmit Descriptor Base Address High—TDBAH [0:3][0:7] (0xE004 + 0x40*n [n=0...7]; R/W)

These registers contain the upper 32 bits of the 64-bit descriptor base address.

Table 28-113.Transmit Descriptor Base Address High—TDBAH [0:3][0:7] (0xE004 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E004h at 0x40 **Offset End:** E007h at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TDBAH	Transmit Descriptor Base Address [63:32]		X	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3804, 0x3904, 0x3A04 & 0x3B04 respectively.

28.11.1.12 Transmit Descriptor Ring Length—TDLEN [0:3][0:7] (0xE008 + 0x40*n [n=0...7]; R/W)

These registers contain the descriptor ring length. The registers indicates the length in bytes and must be 128-byte aligned.

Table 28-114.Transmit Descriptor Ring Length—TDLEN [0:3][0:7] (0xE008 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E008h at 0x40 **Offset End:** E00Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Reserved	Reserved Reads as 0b. Should be written to 0b.			
19 :07	LEN	Descriptor Ring Length (number of 8 descriptor sets)		0x0	R/W
06 :00	0	Ignore on writes. Reads back as 0b.		0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3808, 0x3908, 0x3A08 & 0x3B08 respectively.

28.11.1.13 Transmit Descriptor Head—TDH [0:3][0:7] (0xE010 + 0x40*n [n=0...7]; RO)

These registers contain the head pointer for the transmit descriptor ring. It points to a 16-byte datum. Hardware controls this pointer.

Note: The values in these registers might point to descriptors that are still not in host memory. As a result, the host cannot rely on these values in order to determine which descriptor to release.

Table 28-115.Transmit Descriptor Head—TDH [0:3] [0:7] (0xE010 + 0x40*n [n=0...7]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E010h at 0x40 **Offset End:** E013h at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Should be written to 0b.			
15 :00	TDH	Transmit Descriptor Head		0x0	RO

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3810, 0x3910, 0x3A10 & 0x3B10 respectively.

28.11.1.14 Transmit Descriptor Tail—TDT [0:3][0:7] (0xE018 + 0x40*n [n=0...7]; R/W)

These registers contain the tail pointer for the transmit descriptor ring and points to a 16-byte datum. Software writes the tail pointer to add more descriptors to the transmit ready queue. Hardware attempts to transmit all packets referenced by descriptors between head and tail.

Table 28-116.Transmit Descriptor Tail—TDT [0:3][0:7] (0xE018 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E018h at 0x40 **Offset End:** E01Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Reads as 0b. Should be written to 0b for future compatibility.			
15 :00	TDT	Transmit Descriptor Tail		0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3818, 0x3918, 0x3A18 & 0x3B18 respectively.

28.11.1.15 Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W)

These registers control the fetching and write-back operations of transmit descriptors. The three threshold values are used to determine when descriptors are read from and written to host memory. The values are in units of descriptors (each descriptor is 16 bytes).

Since write-back of transmit descriptors is optional (under the control of *RS* bit in the descriptor), not all processed descriptors are counted with respect to WTHRESH. Descriptors start accumulating after a descriptor with *RS* set is processed. In addition, with transmit descriptor bursting enabled, some descriptors are written back that did not have *RS* set in their respective descriptors.

Note: When WTHRESH = 0b, only descriptors with the *RS* bit set are written back

Table 28-117.Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E028h at 0x40 **Offset End:** E02Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :28	HWBTHRESH	Transmit Head writeback threshold If value of field is greater than 0x0, head writeback to host will occur only when the amount of internal pending write backs exceeds this threshold. See Section 26.2.3 for additional information. **Note:** When activating this mode the *WB_on_EITR* bit in the *TDWBAL* register should be set to guarantee a write back after a timeout even if the threshold has not been reached.		0x0	R/W
26	SWFLSH (WC)	Transmit Software Flush This bit enables software to trigger descriptor write-back flushing, independently of other conditions. This bit is self cleared by hardware. Bit will clear after write-back flush is triggered (may take a number of cycles). **Note:** When working in head write-back mode (*TDWBAL.Head_WB_En* = 1) *TDWBAL.WB_on_EITR* bit should be set for transmit descriptor flush to occur.		0h	R/W
25	ENABLE	Transmit Queue Enable When set, this bit enables the operation of a specific transmit queue. Setting this bit initializes the Tail and Head registers (*TDT*[n] and *TDH*[n]) of a specific queue. Until then, the state of the queue is kept and can be used for debug purposes. When disabling a queue, this bit is cleared only after all transmit activity on this queue is stopped. **Note:** When transmit queue is enabled and descriptors exist, descriptors and data are fetched immediately. Actual transmit activity on port starts only if the *TCTL.EN* bit is set.		0h	R/W
24 :21	Reserved	Reserved			

Table 28-117.Transmit Descriptor Control—TXDCTL [0:3][0:7] (0xE028 + 0x40*n [n=0...7]; R/W) (Sheet 2 of 2)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** E028h at 0x40 **Offset End:** E02Bh at 0x40	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
20 :16	WTHRESH	Write-Back Threshold Controls the write-back of processed transmit descriptors. This threshold refers to the number of transmit descriptors in the on-chip buffer that are ready to be written back to host memory. In the absence of external events (explicit flushes), the write-back occurs only after at least WTHRESH descriptors are available for write-back. Possible values for this field are 0 to 23. **Note:** Since the default value for write-back threshold is 0b, descriptors are normally written back as soon as they are processed. WTHRESH must be written to a non-zero value to take advantage of the write-back bursting capabilities of the Controller.			0x0	R/W
15 :13	Reserved	Reserved				
12 :08	HTHRESH	Host Threshold Prefetch of transmit descriptors is considered when number of valid transmit descriptors in host memory is at least HTHRESH. **Note:** HTHRESH should be given a non zero value each time PTHRESH is used.			0x0	R/W
07 :05	Reserved	Reserved				
04 :00	PTHRESH	Prefetch Threshold Controls when a prefetch of descriptors is considered. This threshold refers to the number of valid, unprocessed transmit descriptors the Controller has in its on-chip buffer. If this number drops below PTHRESH, the algorithm considers pre-fetching descriptors from host memory. However, this fetch does not happen unless there are at least HTHRESH valid descriptors in host memory to fetch. **Note:** When PTHRESH is 0x0 a Transmit descriptor fetch operation is done when any valid descriptors are available in Host memory and space is available in internal buffer.			0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3828, 0x3928, 0x3A28 and 0x3B28 respectively.

28.11.1.16 Tx Descriptor Completion Write-Back Address Low—TDWBAL [0:3][0:7] (0xE038 + 0x40*n [n=0...7]; R/W)

Table 28-118.Tx Descriptor Completion Write-Back Address Low—TDWBAL [0:3][0:7] (0xE038 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E038h at 0x40 **Offset End:** E03Bh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	HeadWB_Low	Lowest 32 bits of the head write-back memory location (DWORD aligned). Last 2 bits of this field are ignored and are always read as 0.0, meaning that the actual address is QWORD aligned.		0x0	R/W
01	WB_on_EITR	When set, a head write back is done upon EITR expiration.		0h	R/W
00	Head_WB_En	Head Write-Back Enable 1b = Head write-back is enabled. 0b = Head write-back is disabled. When head_WB_en is set, SN_WB_en is ignored and no descriptor write-back is executed.		0h	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3838, 0x3938, 0x3A38 & 0x3B38 respectively.

28.11.1.17 Tx Descriptor Completion Write-Back Address High - TDWBAH [0:3][0:7] (0xE03C + 0x40*n [n=0...7];R/W)

Table 28-119.Tx Descriptor Completion Write-Back Address High - TDWBAH [0:3][0:7] (0xE03C + 0x40*n [n=0...7];R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E03Ch at 0x40 **Offset End:** E03Fh at 0x40	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HeadWB_High	Highest 32 bits of the head write-back memory location.		0x0	R/W

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x383C, 0x393C, 0x3A3C & 0x3B3C respectively.

28.12 DCA and TPH Registers

28.12.1 Detailed Register Descriptions

28.12.1.1 Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W)

Note: RX data write no-snoop is activated when the NSE bit is set in the receive descriptor.

Table 28-120.Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** C014h at 0x40 **Offset End:** C017h at 0x40	
Size: 32 bit	**Default:** 0x0000A200			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	CPUID	Physical ID Legacy DCA capable platforms - the device driver, upon discovery of the physical CPU ID and CPU Bus ID, programs the CPUID field with the Physical CPU and Bus ID associated with this Rx queue.		0x0	R/W
23 :16	Reserved	Reserved			
15	RxRepHeader ROEn	Receive Replicated/Split Header Relax Order Enable		1h	R/W
14	RxRepHeader NSEn	Receive Replicated/Split Header No Snoop Enable This bit must be reset to 0b to ensure correct functionality of header write to host memory. Note: When TPH is enabled No Snoop bit should be 0.		0h	R/W
13	RXdataWrite ROEn	Receive Data Write Relax Order Enable (header replication: header and data)		1h	R/W
12	RXdataWrite NSEn	Receive Data Write No Snoop Enable (header replication: header and data) When set to 0b, the last bit of the *Packet Buffer Address* field in the advanced receive descriptor is used as the LSB of the packet buffer address (A0), thus enabling Byte alignment of the buffer. When set to 1b, the last bit of the *Packet Buffer Address* field in advanced receive descriptor is used as the No-Snoop Enabling (NSE) bit (buffer is Word aligned). If also set to 1b, the NSE bit determines whether the data buffer is snooped or not. Note: When TPH is enabled No Snoop bit should be 0.		0h	R/W
11	RXdescW-BROen (RO)	Receive Descriptor Write-Back Relax Order Enable This bit must be reset to 0b to ensure correct functionality of descriptor write-back.		0h	R/W
10	RXdescWBNSen	Receive Descriptor Write-Back No Snoop Enable This bit must be reset to 0b to ensure correct functionality of descriptor write-back. Note: When TPH is enabled No Snoop bit should be 0.		0h	R/W
09	RXdescReadROEn	Receive Descriptor Read Relax Order Enable		1h	R/W

Table 28-120.Rx DCA Control Registers—RXCTL [0:3][0:7] (0xC014 + 0x40*n [n=0...7]; R/W) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** C014h at 0x40 **Offset End:** C017h at 0x40	
Size: 32 bit	**Default:** 0x0000A200			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
08	RXdescReadNSEn	Receive Descriptor Read No Snoop Enable This bit must be reset to 0b to ensure correct functionality (Except if the software driver can guarantee the data is present in the main memory before the DMA process occurs). Note: When TPH is enabled No Snoop bit should be 0.		0h	R/W
07	Rx Payload DCA EN	Receive Payload DCA Enable When set, hardware enables DCA for all Ethernet payloads written into memory. When cleared, hardware does not enable DCA for Ethernet payloads. This bit is cleared as a default.		0h	R/W
06	Rx Header DCA EN	Receive Header DCA Enable When set, hardware enables DCA for all received header buffers. When cleared, hardware does not enable DCA for Rx headers. This bit is cleared as a default.		0h	R/W
05	RX Descriptor DCA EN	Descriptor DCA Enable When set, hardware enables DCA for all Rx descriptors written back into memory. When cleared, hardware does not enable DCA for descriptor write-backs. This bit is cleared as a default.		0h	R/W
04	Reserved	Reserved		0h	R/W
03 :00	Reserved	Reserved			

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x2814, 0x2914, 0x2A14 & 0x2B14 respectively.

28.12.1.2 Tx DCA Control Registers—TXCTL [0:3][0:7] (0xE014 + 0x40*n [n=0...7]; R/W)

Table 28-121.Tx DCA Control Registers—TXCTL [0:3][0:7] (0xE014 + 0x40*n [n=0...7]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** E014h at 0x40 **Offset End:** E017h at 0x40	
Size: 32 bit	**Default:** 0x00002A00			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	CPUID	Physical ID Legacy DCA capable platforms - the device driver, upon discovery of the physical CPU ID and CPU Bus ID, programs the CPUID field with the Physical CPU and Bus ID associated with this Tx queue.		0x0	R/W
23 :14	Reserved	Reserved Write 0 ignore on read.			
13	TXDataReadROEn	Tx Data Read Relax Order Enable		1h	R/W
12	TXDataReadNSEn	Tx Data Read No Snoop Enable		0h	R/W
11	TXdescWBROen	Tx Descriptor Write-Back Relax Order Enable Applies to head write-back, when enabled.		1h	R/W
10	TXdescWBNSen	Tx Descriptor Write-Back No Snoop Enable This bit must be reset to 0b to ensure correct functionality of descriptor write-back. Also applies to head write-back, when enabled.		0h	R/W
09	TXdescRDROEn	Tx Descriptor Read Relax Order Enable		1h	R/W
08	TXdescRDNSen	Tx Descriptor Read No Snoop Enable This bit must be reset to 0b to ensure correct functionality (unless the software device driver has written this bit with a write-through instruction). Note: When TPH is enabled No Snoop bit should be 0.		0h	R/W
07 :06	Reserved	Reserved			
05	TX Descriptor DCA EN	Descriptor DCA Enable When set, hardware enables DCA for all Tx descriptors written back into memory. When cleared, hardware does not enable DCA for descriptor write-backs. This bit is cleared as a default and also applies to head write-back when enabled.		0h	R/W
04 :00	Reserved	Reserved			

Note: To keep compatibility with the 82575, for queues 0-3, these registers are aliased to addresses 0x3814, 0x3914, 0x3A14 & 0x3B14 respectively.

28.12.1.3 DCA Requester ID Information—DCA_ID [0:3] (0x5B70; RO)

Note: The DCA requester ID field, composed of Device ID, Bus #, and Function # is set up in MMIO space for software to program the DCA Requester ID Authentication register.

Table 28-122.DCA Requester ID Information—DCA_ID [0:3] (0x5B70; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B70h **Offset End:** 5B73h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved				
15 :08	Bus Number	Bus Number Bus number assigned to the function based on BIOS/operating system enumeration.			0x0	RO
07 :03	Device Number	Device Number Device number assigned to the function based on BIOS/operating system enumeration.			0x0	RO
02 :00	Function Number	Function Number Function number assigned to the function based on BIOS/operating system enumeration.			0h	RO

28.12.1.4 DCA Control—DCA_CTRL [0:3] (0x5B74; R/W)

This CSR is common to all functions.

Table 28-123.DCA Control—DCA_CTRL [0:3] (0x5B74; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5B74h **Offset End:** 5B77h	
Size: 32 bit	**Default:** 0x1				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved				
04 :01	DCA_MODE	DCA Mode 000b = Legacy DCA is supported. The TAG field in the TLP header is based on the following coding: bit 0 is DCA enable; bits 3:1 are CPU ID). 001b - 111b = Reserved			0x0	R/W
00	DCA_DIS	DCA Disable 0b = DCA tagging is enabled. 1b = DCA tagging is disabled.			1h	R/W

28.13 Virtualization Registers

28.13.1 Detailed Register Descriptions

28.13.1.1 VMDq Control Register—VT_CTL [0:3] (0x581C; R/W)

Table 28-124.VMDq Control Register—VT_CTL [0:3] (0x581C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 581Ch **Offset End:** 581Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved			
30	Rpl_En	Replication Enable		0h	R/W
29	Dis_Def_Pool	Drop if no pool is found. If this bit is asserted, then in a RX switching virtualized environment, if there is no destination pool, the packet is discarded and not sent to the default pool. Otherwise, it is sent to the pool defined by the DEF_PL field.		0h	R/W
28	IGMAC	If set, MAC address is ignored during pool decision. Pooling is based on VLAN only. If this bit is set, then the *VMOLR.strvlan* should be set to the same value for all pools,		0h	R/W
27	FLP	Filter local packets - filter incoming packets whose MAC source address matches one of the LAN port DA MAC addresses. If the SA of the received packet matches one of the DA in the RAH/RAL registers, then the VM tied to this DA does not receive the packet. Other VMs can still receive it.		0h	R/W
26 :10	Reserved	Reserved			
09 :07	DEF_PL	Default pool - used to queue packets that did not pass any VM queuing decision.		000h	R/W
06 :00	Reserved	Reserved			

28.13.1.2 Malicious Driver Free Block—MDFB [0:3] (0x3558; RWS)

Table 28-125.Malicious Driver Free Block—MDFB [0:3] (0x3558; RWS)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3558h **Offset End:** 355Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07 :00	Block Queue	Indicates queue that was blocked due to malicious behavior. When bit is set, to commence activity on offending queue, Software should initiate a software reset and re-initialize all queues on the port.		0x0	RWS

28.13.1.3 Last VM Misbehavior Cause—LVMMC [0:3] (0x3548; RC)

Bits in LVMMC register define the cause for blocking the malicious queue that was reported in the *MDFB.Block Queue* field.

Table 28-126. Last VM Misbehavior Cause—LVMMC[0:3] (0x3548; RC) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3548h **Offset End:** 354Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Last_Q	Last queue that detected malicious behavior.		0x0	RC
28	Mal_PF	Malicious Driver behavior detected on current PF		0h	RC
27 :25	Reserved	Reserved Ignore on read write 0.			
24	ILL_DBU	Illegal DBU configuration.		0h	RC
23	Reserved	Reserved			
22	No EOP	Packet without EOP (for example, bigger than the ring size)		0h	RC
21	Wrong_null	Null without EOP		0h	RC
20	DESC_TYPE	Wrong descriptor type (other than 2,3)		0h	RC
19	Reserved	Reserved			
18	SSO_TCP	Wrong parameter of headers for TCP SSO		0h	RC
17	SSO_UDP	Wrong parameter of headers for UDP SSO		0h	RC
16 :12	Reserved	Reserved			
11	SCTP_aligned	CRC request of non 4 byte aligned data.		0h	RC
10	Off_Ill	Illegal offload request.		0h	RC

Table 28-126. Last VM Misbehavior Cause—LVMMC[0:3] (0x3548; RC) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 3548h **Offset End:** 354Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
09	Adv_Size	Illegal advanced descriptor size.		0h	RC
08	Leg_Size	Illegal legacy descriptor size.		0h	RC
07	SCTP SSO	Illegal SCTP header was detected in a single send operation.		0h	RC
06	UDP LSO	Illegal UDP header was detected in a large send operation.		0h	RC
05	Reserved	Reserved			
04	TCP LSO	Illegal TCP header was detected in a large send operation.		0h	RC
03	Wrong MAC_IP	Wrong MAC+IP header size		0h	RC
02	IPV6 Header	Illegal IPV6 header size.		0h	RC
01	IPV4 Header	Illegal IPV4 header size.		0h	RC
00	Mac Header	Illegal MAC header size.		0h	RC

28.13.1.4 VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW)

This register controls the offload and queueing options applied to each pool (VM).

Table 28-127.VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5AD0h at 0x4 **Offset End:** 5AD3h at 0x4	
Size: 32 bit	**Default:** 0x80002600			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved - must be set to one.			
30	strvlan	VLAN strip		0h	RW
29	Reserved	Reserved			
28	mpe	multicast promiscuous		0h	RW
27	bam	Broadcast accept		0h	RW
26	rope	receive overflow packets - accept packets that match the UTA table.		0h	RW
25	rompe	receive overflow multicast packets - accept packets that match the MTA table.		0h	RW

Table 28-127.VM Offload Register—VMOLR [0:3][0:7] (0x5AD0 + 4*n [n=0...7]; RW) (Sheet 2 of 2)

Description:					
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1	Offset Start: 5AD0h at 0x4 Offset End: 5AD3h at 0x4	
Size: 32 bit	Default: 0x80002600			Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
24	aupe	Accept Untagged packets enable. When set, packets without VLAN tag can be forwarded to this queue, assuming they pass the MAC address queueing mechanism.		0h	RW
23: 17	Reserved	Reserved			
16	lpe	Long packet enable		0h	RW
15 :14	Reserved	Reserved			
13 :00	rlpml	Long packet size (9k default)		2600h	RW

28.13.1.5 Replication Offload Register—RPLOLR [0:3] (0x5AF0; RW)

This register describes the off loads applied to multicast packets.

Table 28-128.Replication Offload Register—RPLOLR [0:3] (0x5AF0; RW

Description:					
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1	Offset Start: 5AF0h Offset End: 5AF3h	
Size: 32 bit	Default: 0x80000000			Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved - must be set to one.			
30	strvlan	VLAN strip		0h	RW
29 :00	Reserved	Reserved			

28.13.1.6 VLAN VM Filter—VLVF [0:3][0:31] (0x5D00 + 4*n [n=0...31]; RW)

This register set describes which VLANs the local VMs are part of. Each of the 32 registers contains a VLAN tag and a list of the VMs which are part of it. Only packets with a VLAN matching one of the VLAN tags of which the VM is member of are forwarded to this VM

Table 28-129.VLAN VM Filter—VLVF [0:3][0:31] (0x5D00 + 4*n [n=0...31]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DD0h at 0x4 **Offset End:** 5DD3h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	VI_En	VLAN Id Enable - this filter is valid. Note: If *RCTL.VFE* is 0 all *VLVF* filters are disabled.		0h	RW
30 :20	Reserved	Reserved Write 0, ignore on read.			
19 :12	POOLSEL	Pool Select (bitmap) Field defines to which VMs a packet with the VLAN_Id should be forwarded to. A bit is allocated to each of the 8 VMs, enabling forwarding the packet with the VLAN_Id to multiple VMs.		0x0	RW
11 :00	VLAN_Id	Defines a VLAN tag, to which each VM whose bit is set in the POOLSEL field, belongs to.		0x0	RW

28.13.1.7 Unicast Table Array—UTA [0:3][0:127] (0xA000 + 4*n [n=0...127]; R/W)

There is one register per 32 bits of the Unicast Address Table for a total of 128 registers (the UTA[127:0] designation). Software must mask to the desired bit on reads and supply a 32-bit word on writes. The first bit of the address used to access the table is set according to the *RCTL.MO* field.

Note: All accesses to this table must be 32 bit.

The lookup algorithm is the same one used for the MTA table.

This table should be zeroed by software before start of work.

Table 28-130.Unicast Table Array—UTA [0:3][0:127] (0xA000 + 4*n [n=0...127]; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** A000h at 0x4 **Offset End:** A003h at 0x4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Bit Vector	Word wide bit vector specifying 32 bits in the unicast destination address filter table.		X	R/W

28.13.1.8 Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW)

Table 28-131.Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DB0h **Offset End:** 5DB3h	
Size: 32 bit	**Default:** 0x00000800			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	Reserved	Reserved			
17 :08	INTERVAL	BSC/MSC Time-interval-specification: The interval size for applying Ingress Broadcast or Multicast Storm Control. Interrupt decisions are made at the end of each interval (and most flags are also set at interval end). Setting this field resets the counter.		0x8	RW
07 :05	Reserved	Reserved			
04	BIDU	BSC Includes Destination Unresolved packets: If bit is set, unicast received packets with no destination pool and sent to the default pool are included in IBSC		0h	RW
03	BDICW	Drop broadcast packets (excluding flow control and manageability packets) if broadcast threshold is exceeded in current window		0h	RW

Table 28-131.Storm Control Control Register—SCCRL [0:3] (0x5DB0; RW) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DB0h **Offset End:** 5DB3h	
Size: 32 bit	**Default:** 0x00000800			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BDIPW	Drop broadcast packets (excluding flow control and manageability packets) if broadcast threshold is exceeded in previous window		0h	RW
01	MDICW	Drop multicast packets (excluding flow control and manageability packets) if multicast threshold is exceeded in current window		0h	RW
00	MDIPW	Drop multicast packets (excluding flow control and manageability packets) if multicast threshold is exceeded in previous window		0h	RW

28.13.1.9 Storm Control Status—SCSTS [0:3] (0x5DB4;RO)

Table 28-132.Storm Control Status—SCSTS [0:3] (0x5DB4;RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DB4h **Offset End:** 5DB7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03	MSCAP	Multicast storm control active in previous window		0h	RO
02	MSCA	Multicast storm control active		0h	RO
01	BSCAP	Broadcast storm control active in previous window		0h	RO
00	BSCA	Broadcast storm control active		0h	RO

28.13.1.10 Broadcast Storm Control Threshold—BSCTRH [0:3] (0x5DB8;RW)

Table 28-133.Broadcast Storm Control Threshold—BSCTRH [0:3] (0x5DB8;RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DB8h **Offset End:** 5DBBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	Reserved	Reserved			
18 :00	UTRESH	Traffic Upper Threshold-size: Represents the upper threshold for broadcast storm control.		0x0	RW

28.13.1.11 Multicast Storm Control Threshold—MSCTRH [0:3] (0x5DBC; RW)

Table 28-134.Multicast Storm Control Threshold—MSCTRH [0:3] (0x5DBC; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DBCh **Offset End:** 5DBFh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :19	Reserved	Reserved			
18 :00	UTRESH	Traffic Upper Threshold-size: Represents the upper threshold for multicast storm control.		0x0	RW

28.13.1.12 Broadcast Storm Control Current Count - BSCCNT [0:3] (0x5DC0;RO)

Table 28-135.Broadcast Storm Control Current Count - BSCCNT [0:3] (0x5DC0;RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5DC0h **Offset End:** 5DC3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved			
24 :00	CCOUNT	IBSC Traffic Current Count: Represents the count of broadcast traffic received in the current time interval in units of 64-byte segments.		0x0	RO

28.13.1.13 Multicast Storm Control Current Count—MSCCNT [0:3] (0x5DC4;RO)

Table 28-136.Multicast Storm Control Current Count—MSCCNT [0:3] (0x5DC4;RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5DC4h **Offset End:** 5DC7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved.			
24 :00	CCOUNT	IMSC Traffic Current Count: Represents the count of multicast traffic received in the current time interval in units of f 64-byte segments.		0x0	RO

28.13.1.14 Storm Control Time Counter—SCTC [0:3] (0x5DC8; RO)

Table 28-137.Storm Control Time Counter—SCTC [0:3] (0x5DC8; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5DC8h **Offset End:** 5DCBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :10	Reserved	Reserved			
09 :00	COUNT	SC Time Counter: The counter for number of time units elapsed since the end of the last time interval.		0x0	RO

This register keeps track of the number of time units elapsed since the end of last time interval.

28.13.1.15 Storm Control Basic Interval—SCBI [0:3] (0x5DCC; RW)

This register defines the basic interval used as the base for the SCCRL.Interval counting in 10 Mb/s speed. This register is defined in 16 ns clock cycles. The interval in 1000/100 is 100 or 10 time smaller respectively.

Table 28-138.Storm Control Basic Interval—SCBI [0:3] (0x5DCC; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5DCCh **Offset End:** 5DCFh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved				
24 :00	BI	Basic interval			0x5F5E10	RW

28.13.1.16 Virtual Mirror Rule Control—VMRCTL [0:3][0:3] (0x5D80 + 0x4*n [n= 0...3]; RW)

This register controls the rules to be applied and the destination port.

Table 28-139.Virtual Mirror Rule Control—VMRCTL [0:3][0:3] (0x5D80 + 0x4*n [n= 0...3]; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5D80h at 0x4 **Offset End:** 5D83h at 0x4	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :11	Reserved	Reserved				
10 :08	MP	VM Mirror port destination. Packets destined to certain VLAN groups, are mirrored to the queue defined by the MP field, according to the *VMRVLAN* register. Packets destined to certain VMs, are mirrored to the queue defined by the MP field, according to the *VMRVM* register. Note: If the VMRCTL.UPME bit is set to 1 all packets received on the port will be forwarded to the queue defined in the MP field.			0x0	RW
07 :04	Reserved	Reserved				
03	VLME	VLAN mirroring enable. Reflects all the traffic received in a set of given VLANs. bit enables mirroring operation based			0h	RW
02	Reserved	Reserved				
01	UPME	Uplink port mirroring enable. Reflects all the traffic received from the network.			0h	RW
00	VPME	Virtual pool mirroring enable. Reflects all the packets sent to a set of given VMs.			0h	RW

28.13.1.17 Virtual Mirror Rule VLAN—VMRVLAN [0:3][0:3] (0x5D90 + 0x4*n [n= 0...3]; RW)

This register controls the VLAN ports as listed in the VLVF table taking part in the VLAN mirror rule.

Table 28-140.Virtual Mirror Rule VLAN—VMRVLAN [0:3][0:3](0x5D90 + 0x4*n [n= 0...3]; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5D90h at 0x4 **Offset End:** 5D93h at 0x4	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VLAN	Bitmap listing that defines which of the 32 VLANs defined in the *VLVF* registers participate in the mirror rule. Packets that have a matching Vlan_ID as defined by the VLVF registers will also be forwarded (mirrored) to the queue defined in the *VMRCTL.MP* field, if the *VMRCTL.VLME bit* is set to 1.			0x0	RW

28.13.1.18 Virtual Mirror Rule VM—VMRVM [0:3][0:3] (0x5DA0 + 0x4*n [n= 0...3]; RW)

This register controls the VLAN ports as listed in the VLVF table taking part in the VLAN mirror rule.

Table 28-141.Virtual Mirror Rule VM—VMRVM [0:3][0:3] (0x5DA0 + 0x4*n [n= 0...3]; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5DA0h at 0x4 **Offset End:** 5DA3h at 0x4	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved				
07 :00	VM	Bitmap listing of VMs participating in the mirror rule. Packets that are forwarded to the queues defined in the VMRVM.VM field, will also be forwarded (mirrored) to the queue defined in the *VMRCTL.MP* field, if the *VMRCTL.VPME bit* is set to 1.			0x0	RW

28.14 Power Management Registers

The following registers enable control of various DMA power saving features.

28.14.1 Detailed Register Descriptions

28.14.1.1 DMA Receive Power Saving Register—DMARPS [0:3] (0x2500; R/W)

Table 28-142.DMA Receive Power Saving Register—DMARPS [0:3] (0x2500; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2500h **Offset End:** 2503h	
Size: 32 bit	**Default:** 0x0000010A			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved Write 0 ignore on read.			
15 :08	RXLPKBL	Receive low power low Kilobyte threshold This value indicates maximum receive buffer full threshold that enables DMA to move to low power mode. Value in 1KB blocks. Maximum allowed value depends on Receive buffer size (IRPBS.RXPbsize). Note: RXBLPL<= RXBLPH value.		0x1	R/W
07 :00	RXLPKBH	Receive low power high Kilobyte threshold This value indicates minimum receive buffer full threshold that causes DMA to move to high power mode. Value in 1KB blocks. Maximum allowed value depends on Receive buffer size (IRPBS.RXPbsize) Note: When RXBLPH Value is 0x0, receive Kilobyte threshold is not used to decide DMA move between high and low power modes.		0xA	R/W

28.14.1.2 DMA Transmit Power Saving Register—DMATPS [0:3] (0x2504; R/W)

Table 28-143.DMA Transmit Power Saving Register—DMATPS [0:3] (0x2504; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 2504h **Offset End:** 2507h	
Size: 32 bit	**Default:** 0x0000010A			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	DMAPS_EN	DMA Power Saving Enable. Bit enables DMA power saving as a function of utilization. 0 - Disable DMA power saving 1 - Enable DMA power saving		0h	R/W
30 :15	Reserved	Reserved Write 0 ignore on read.			
14 :08	TXLPKBL	Transmit low power low Kilobyte threshold This value indicates maximum transmit buffer full threshold that causes DMA to move to high power mode. Value in 1KB blocks. Maximum allowed value depends on Transmit buffer size (*ITPBS.TXPbsize*). Note: TXBLPL<= TXBLPH value.		0x1	R/W
07	Reserved	Reserved Write 0 ignore on read.			
06 :00	TXLPKBH[1]	Transmit low power high Kilobyte threshold This value indicates minimum transmit buffer full threshold that enables DMA to move to low power mode. Value in 1KB blocks. Maximum allowed value depends on Transmit buffer size (ITPBS.TXPbsize). Note: When TXBLPH Value is 0x0, transmit Kilobyte threshold is not used to decide DMA move between high and low power modes.		0xA	R/W

1. Value of TXLPKBH should be greater than maximum packet size to minimize latency.

28.15 Timer Registers

28.15.1 Detailed Register Descriptions

28.15.1.1 Watchdog Setup—WDSTP [0:3] (0x1040; R/W)

Table 28-144.Watchdog Setup—WDSTP [0:3] (0x1040; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1040h **Offset End:** 1043h	
Size: 32 bit	**Default:** 0x01000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	WD_Timeout	Defines the number of seconds until the watchdog expires. The granularity of this timer is 1 sec. The minimal value allowed for this register when the watchdog mechanism is enabled is two. Setting this field to 1b might cause the watchdog to expire immediately. Note: Only 4 LSB bits loaded from EEPROM. Initial value of 4 MSB bits is 0000b.		0x01	R/W
23 :16	WD_Timer (RWS)	Indicates the current value of the timer. Resets to the timeout value each time the Controller functional bit in Software Device Status register is set. If this timer expires, the WD interrupt to the firmware and the WD SDP is asserted. As a result, this timer is stuck at zero until it is re-armed. **Note:** Writing to this field is only for DFX purposes.		X	R/W
15 :02	Reserved	Reserved			
01	WD_Timer_ Load_enable (SC)	Enables the load of the watchdog timer by writing to WD_Timer field. If this bit is not set, the WD_Timer field is loaded by the value of WD_Timeout. **Note:** Writing to this field is only for DFX purposes.		0h	R/W
00	WD_Enable	Enable Watchdog Timer		0h[1]	R/W

1. Value read from the EEPROM.

28.15.1.2 Watchdog Software Device Status—WDSWSTS [0:3] (0x1044; R/W)

Table 28-145.Watchdog Software Device Status—WDSWSTS [0:3] (0x1044; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1044h **Offset End:** 1047h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Stuck Reason	This field can be used by software to indicate to the firmware the reason the Controller is malfunctioning. The encoding of this field is software/firmware dependent. A value of 0 indicates a functional Controller.			0x0	R/W
23 :02	Reserved	Reserved				
01	Force_WD (SC)	Setting this bit causes the WD timer to expire immediately. The WD_timer field is set to 0b. It can be used by software in order to indicate some fatal error detected in the software or in the hardware. This bit is self clearing.			0h	R/W
00	Dev_Functional (SC)	Each time this bit is set, the watchdog timer is re-armed. This bit is self clearing			0h	R/W

28.15.1.3 Free Running Timer—FRTIMER [0:3] (0x1048; RWS)

This register reflects the value of a free running timer that can be used for various timeout indications. The register is reset by a PCI reset and/or software reset.

Note: Writing to this register is for DFX purposes only.

Table 28-146.Free Running Timer—FRTIMER [0:3] (0x1048; RWS)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1048h **Offset End:** 104Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	Seconds	Number of seconds from the timer start (up to 4095 seconds).			X	RWS
19 :10	Millisecond	Number of milliseconds in the current second.			X	RWS
09 :00	Microsecond	Number of microseconds in the current millisecond.			X	RWS

28.15.1.4 TCP Timer—TCPTIMER [0:3] (0x104C; R/W)

Table 28-147.TCP Timer—TCPTIMER [0:3] (0x104C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 104Ch **Offset End:** 104Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11	Loop	TCP Loop When set to 1b, the TCP counter reloads duration each time it reaches zero, and continues down-counting from this point without kick-starting. When set to 0b, the TCP counter stops at a zero value and does not re-start until KickStart is activated. **Note:** Setting this bit alone is not enough to start the timer activity. The KickStart bit should also be set.		0h	R/W
10	TCPCountFinish (WS)	TCP Count Finish This bit enables software to trigger a TCP timer interrupt, regardless of the internal state. Writing a 1b to this bit triggers an interrupt and resets the internal counter to its initial value. Down-count does not restart until either KickStart is activated or Loop is set. Writing a 0b has no effect.		0h	R/W
09	TCPCountEn	TCP Count Enable 1b = TCP timer counting enabled. 0b = TCP timer counting disabled. Once enabled, the TCP counter counts from its internal state. If the internal state is equal to 0b, the down-count does not restart until KickStart is activated. If the internal state is not 0b, the down-count continues from internal state. This enables a pause in the counting for debug purpose.		0h	R/W
08	KickStart (WS)	Counter Kick-Start Writing a 1b to this bit kick-starts the counter down-count from the initial value defined in the *Duration* field. Writing a 0b has no effect.		0h	R/W
07 :00	Duration	Duration Duration of the TCP interrupt interval in msec.		0x0	R/W

28.16 Time Sync Register Descriptions

28.16.1 Detailed Register Descriptions

28.16.1.1 RX Time Sync Control Register—TSYNCRXCTL [0:3] (0xB620;RW)

Table 28-148.RX Time Sync Control Register—TSYNCRXCTL [0:3] (0xB620;RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B620h **Offset End:** B623h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04	En	Enable RX timestamp 0 = time stamping disabled. 1 = time stamping enabled.		0h	RW
03 :01	Type	Type of packets to timestamp - 000b – time stamp L2 (V2) packets only (Sync or Delay_req depends on message type in Section 28.16.1.26 and packets with Pdelay_Req and Pdelay_Resp message ID values) 001b – time stamp L4 (V1) packets only (Sync or Delay_req depends on message type in Section 28.16.1.26) 010b – time stamp V2 (L2 and L4) packets (Sync or Delay_req depends on message type in Section 28.16.1.26 and packets with Pdelay_Req and Pdelay_Resp message ID values) 100b – time stamp all packets. In this mode no locking is done to the timestamp value in the *RXSTMPL/H* timestamp registers, the RDESC.STATUS.TS bit in the receive descriptor stays 0, while the RDESC.STATUS.TSIP bit in the receive descriptor is always 1 if placing timestamp in receive buffer is enabled (See Section 26.1.10). 101b - Time stamp all packets which have a Message Type bit 3 zero, which means timestamp all event packets. This is applicable for V2 packets only. 011b, 110b and 111b – reserved Note: Field is also used for defining packets that have timestamp captured in receive buffer (See Section 26.1.10).		0x0	RW
00	RXTT(RO/V)	Rx timestamp valid (='1' when a valid value for Rx timestamp is captured in the Rx timestamp register, clear by read of Rx timestamp register RXSATRH)		0x0	RW

28.16.1.2 RX timestamp Low—RXSTMPL [0:3] (0xB624; RO)

Table 28-149.RX timestamp Low—RXSTMPL [0:3] (0xB624; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B624h **Offset End:** B627h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RTSL	Rx timestamp LSB value Value in 1 nS resolution.			0x0	RO

28.16.1.3 RX Timestamp High—RXSTMPH [0:3] (0xB628; RO)

Table 28-150.RX Timestamp High—RXSTMPH [0:3] (0xB628; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B628h **Offset End:** B62Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved.				
07 :00	RTSH	Rx timestamp MSB value Value in 2^{32} nS resolution.			0x0	RO

28.16.1.4 RX Timestamp Attributes Low—RXSATRL[0:3] (0xB62C; RO)

Table 28-151.RX Timestamp Attributes Low—RXSATRL[0:3] (0xB62C; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B62Ch **Offset End:** B62Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SourceIDL	Sourceuuid low			0x0	RO

28.16.1.5 RX timestamp Attributes High—RXSATRH [0:3] (0xB630; RO)

Table 28-152.RX timestamp Attributes High—RXSATRH [0:3] (0xB630; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** B630h **Offset End:** B633h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	SequenceID	SequenceId		0x0	RO
15 :00	SourceIDH	Sourceuuid high		0x0	RO

28.16.1.6 TX Time Sync Control Register—TSYNCTXCTL [0:3] (0xB614; RW)

Table 28-153.TX Time Sync Control Register—TSYNCTXCTL [0:3] (0xB614; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** B614h **Offset End:** B617h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :05	Reserved	Reserved			
04	EN	Enable Transmit timestamp 0b = time stamping disabled. 1b = time stamping enabled.		0h	RW
03 :01	Reserved	Reserved			
00	TXTT(RO/V)	Transmit timestamp valid (equals 1b when a valid value for Tx timestamp is captured in the Tx timestamp register, clear by read of Tx timestamp register TXSTMPH)		0h	RW

28.16.1.7 TX Timestamp Value Low—TXSTMPL [0:3] (0xB618;RO)

Table 28-154.TX Timestamp Value Low—TXSTMPL [0:3] (0xB618;RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B618h **Offset End:** B61Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TTSL	Transmit timestamp LSB value Value in 1 nS resolution.		0x0	RO

28.16.1.8 TX Timestamp Value High—TXSTMPH[0:3] (0xB61C; RO)

Table 28-155.TX Timestamp Value High—TXSTMPH[0:3] (0xB61C; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B61Ch **Offset End:** B61Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.			
07 :00	TTSH	Transmit timestamp MSB value Value in 2^{32} nS resolution.		0x0	RO

28.16.1.9 System Time Register Residue—SYSTIMR [0:3] (0xB6F8; RW)

Table 28-156.System Time Register Residue—SYSTIMR [0:3] (0xB6F8; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B6F8h **Offset End:** B6FBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	STR	System time Residue value. Value in 2^{-32} nS resolution.		0x0	RW

28.16.1.10 System Time Register Low—SYSTIML [0:3] (0xB600; RW)

Table 28-157.System Time Register Low—SYSTIML [0:3] (0xB600; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B600h **Offset End:** B603h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	STL	System time LSB value Value in 1 nS resolution.			0x0	RW

28.16.1.11 System Time Register High—SYSTIMH [0:3] (0xB604; RW)

Table 28-158.System Time Register High—SYSTIMH [0:3] (0xB604; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B604h **Offset End:** B607h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.				
07 :00	STH	System time MSB value Value in 2^{32} nS resolution.			0x0	RW

28.16.1.12 Increment Attributes Register—TIMINCA [0:3] (0xB608; RW)

28.16.1.13 Time Adjustment Offset Register Low—TIMADJL [0:3] (0xB60C; RW)

Table 28-159.Time Adjustment Offset Register Low—TIMADJL [0:3] (0xB60C; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B60Ch **Offset End:** B60Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TADJL	Time adjustment value – Low Value in 1 nS resolution.			0x0	RW

28.16.1.14 Time Adjustment Offset Register High—TIMADJH [0:3] (0xB610;RW)

Table 28-160.Time Adjustment Offset Register High—TIMADJH [0:3] (0xB610;RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B610h **Offset End:** B613h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31	Sign	Sign (0b=”+”, 1b =”-“)			0h	RW
30 :08	Reserved	Reserved. Write 0 ignore on read.				
07 :00	TADJH	Time adjustment value - High Value in 2^{32} resolution.			0x0	RW

28.16.1.15 TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW)

Table 28-161.TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B640h **Offset End:** B643h	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Disable systime	Disable SYSTIM count operation 0b - SYSTIM timer activated 1b - SYSTIM timer disabled. Value of SYSTIMH, SYSTIML and SYSTIMR remains constant.		1h	RW
30 :21	Reserved	Reserved			
20	PLSNeg1	Generate Negative pulse on Target Time 1 when PLSG1 is 1. 0 – Generate positive pulse on Target Time 1when PLSG1 is 1. 1 – Generate Negative pulse on target time 1 when PLSG1 is 1. Note: If PLSNeg1 = 1, at start the selected SDP pin is set to 1.		0h	RW
19	PLSG1	Use Target Time 1 to generate start of pulse and Target Time 0 to generate end of pulse. SDP pin selected to drive pulse or level change is set according to the *TSSDP.TS_SDPx_SEL* field with a value of 01b and *TSSDP.TS_SDPx_EN* bit with a value of 1b. 0 – Target Time 1 generates change in SDP level. 1 – Target time 1 generates start of pulse on SDP pin. Note: Pulse or level change is generated when TSAUXC.EN_TT1 is set to 1.		0h	RW
18	PLSNeg0	Generate Negative pulse on Target Time 0 when PLSG0 is 1. 0 – Generate positive pulse on Target Time 0 when PLSG0 is 1. 1 – Generate Negative pulse on target time 0 when PLSG0 is 1. Note: If PLSNeg0 = 1, at start the selected SDP pin is set to 1.		0h	RW
17	PLSG0	Use Target Time 0 to generate start of pulse and Target Time 1 to generate end of pulse. SDP pin selected to drive pulse or level change is set according to the *TSSDP.TS_SDPx_SEL* field with a value of 00h and *TSSDP.TS_SDPx_EN* bit with a value of 1b. 0 – Target Time 0 generates change in SDP level. 1 – Target time 0 generates start of pulse on SDP pin. Note: Pulse or level change is generated when *TSAUXC.EN_TT0* is set to 1.		0h	RW
16 :12	Reserved	Reserved			
11	AUTT1	Auxiliary timestamp taken - cleared when read from auxiliary timestamp 1 occurred		0h	RW
10	EN_TS1	Enable hardware time stamp 1 Enable Time stamping occurrence of change in SDP pin into the *AUXSTMPL*1 and *AUXSTMPH1* registers. SDP pin (0 to 1) is selected for time stamping, if the SDP pin is selected via the *TSSDP.AUX1_SDP_SEL* field and the *TSSDP.AUX1_TS_SDP_EN* bit is set to 1b.		0h	RW
09	AUTT0	Auxiliary timestamp taken - cleared when read from auxiliary timestamp 0 occurred		0h	RW

Table 28-161.TimeSync Auxiliary Control Register—TSAUXC [0:3] (0xB640; RW) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** B640h **Offset End:** B643h	
Size: 32 bit	**Default:** 0x80000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
08	EN_TS0	Enable hardware time stamp 0 Enable Time stamping occurrence of change in SDP pin into the *AUXSTMPL0* and *AUXSTMPH0* registers. SDP pin (0 to 1) is selected for time stamping, if the SDP pin is selected via the *TSSDP.AUX0_SDP_SEL* field and the *TSSDP.AUX0_TS_SDP_EN* bit is set to 1b.		0h	RW
07	ST1	Start Clock 1 Toggle on Target Time 1 Enable Clock 1 toggle only after Target Time 1, that's defined in the *TRGTTIML1* and *TRGTTIMH1* registers, has passed. The clock output is initially 1 and toggles with a frequency defined in the *FREQOUT1* register		0h	RW
06	Reserved	Reserved			
05	EN_CLK1	Enable Configurable Frequency Clock 1 Clock is generated according to frequency defined in the *FREQOUT1* register on the SDP pin (0 to 1) that has both: 1. *TSSDP.TS_SDPx_SEL* field with a value of 11b. 2. *TSSDP.TS_SDPx_EN* value of 1b.		0h	RW
04	ST0	Start Clock 0 Toggle on Target Time 0 Enable Clock 0 toggle only after target time 0, that's defined in the TRGTTIML0 and TRGTTIMH0 registers, has passed. The clock output is initially 0 and toggles with a frequency defined in the FREQOUT0 register.		0h	RW
03	Reserved	Reserved			
02	EN_CLK0	Enable Configurable Frequency Clock 0 Clock is generated according to frequency defined in the *FREQOUT0* register on the SDP pin (0 to 1) that has both: 1. *TSSDP.TS_SDPx_SEL* field with a value of 10b. 2. *TSSDP.TS_SDPx_EN* value of 1b.		0h	RW
01	EN_TT1	Enable target time 1. Enable bit is set by software to 1b, to enable pulse or level change generation as a function of the *TSAUXC.PLSG1* and *TSAUXC.PLSNeg1* bits. The bit is cleared by hardware when the target time is hit and Pulse or level change occurs.		0h	RW
00	EN_TT0	Enable target time 0. Enable bit is set by software to 1b, to enable pulse or level change generation as a function of the *TSAUXC.PLSG0* and *TSAUXC.PLSNeg0* bits. The bit is cleared by hardware when the target time is hit and Pulse or level change occurs.		0h	RW

28.16.1.16 Target Time Register 0 Low—TRGTTIML0 [0:3] (0xB644; RW)

Table 28-162.Target Time Register 0 Low—TRGTTIML0 [0:3] (0xB644; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B644h **Offset End:** B647h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TTL	Target Time 0 LSB register Value in 1 nS resolution.		0x0	RW

28.16.1.17 Target Time Register 0 High—TRGTTIMH0 [0:3] (0xB648; RW)

Table 28-163.Target Time Register 0 High—TRGTTIMH0 [0:3] (0xB648; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B648h **Offset End:** B64Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.			
07 :00	TTH	Target Time 0 MSB register Value in 2^{32} nS resolution.		0x0	RW

28.16.1.18 Target Time Register 1 Low—TRGTTIML1 [0:3] (0xB64C; RW)

Table 28-164.Target Time Register 1 Low—TRGTTIML1 [0:3] (0xB64C; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B64Ch **Offset End:** B64Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TTL	Target Time 1 LSB register Value in 1 nS resolution.		0x0	RW

28.16.1.19 Target Time Register 1 High—TRGTTIMH1 [0:3] (0xB650; RW)

Table 28-165.Target Time Register 1 High—TRGTTIMH1 [0:3] (0xB650; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B650h **Offset End:** B653h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.			
07 :00	TTH	Target Time 1 MSB register Value in 2^{32} nS resolution.		0x0	RW

28.16.1.20 Frequency Out 0 Control Register—FREQOUT0 [0:3] (0xB654; RW)

Table 28-166.Frequency Out 0 Control Register—FREQOUT0 [0:3] (0xB654; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B654h **Offset End:** B657h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.			
07 :00	CHCT	Clock Out Half Cycle Time Half Cycle time of Clock 0 in 8 nS resolution. Clock is generated on SDP pin when T*SAUXC.EN_CLK0* is set to 1. SDP pin (0 to 1) that drives Clock 0 is selected according to the *TSSDP.TS_SDPx_SEL* field that has a value of 10b and a *TSSDP.TS_SDPx_EN* value of 1b. If T*SAUXC.*ST0 is set to 1, start of clock toggle is defined by Target Time 0 (*TRGTTIML0* and *TRGTTIMH0*) registers. Notes: 1. Setting this register to zero while using the frequency out feature, is illegal. 2. Clock 0 generation should not be enabled so long as absolute value of SYSTIM error correction using the TRGTTIMH0 and TRGTTIML0 registers is greater then 2 µsec.		0x0	RW

28.16.1.21 Frequency Out 1 Control Register—FREQOUT1 [0:3] (0xB658; RW)

Table 28-167.Frequency Out 1 Control Register—FREQOUT1 [0:3] (0xB658; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B658h **Offset End:** B65Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.				
07 :00	CHCT	Clock Out Half Cycle Time Half Cycle time of Clock 1 in 8 nS resolution. Clock is generated on SDP pin when T*SAUXC.EN_CLK1* is set to 1. SDP pin (0 to 1) that drives Clock 1 is selected according to the *TSSDP.TS_SDPx_SEL* field that has a value of 11b and a *TSSDP.TS_SDPx_EN* value of 1b. If T*SAUXC.*ST1 is set to 1, start of clock toggle is defined by Target Time 1 (*TRGTTIML1* and *TRGTTIMH1*) registers. Notes: 1. Setting this register to zero while using the frequency out feature, is illegal. 2. Clock 1 generation should not be enabled so long as absolute value of SYSTIM error correction using the TRGTTIMH1 and TRGTTIML1 registers is greater then 2 µsec.			0x0	RW

28.16.1.22 Auxiliary Time Stamp 0 Register Low—AUXSTMPL0 [0:3] (0xB65C; RO)

Table 28-168.Auxiliary Time Stamp 0 Register Low—AUXSTMPL0 [0:3] (0xB65C; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B65Ch **Offset End:** B65Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TSTL	Auxiliary Time Stamp 0 LSB value Value in 1 nS resolution.			0x0	RO

28.16.1.23 Auxiliary Time Stamp 0 Register High—AUXSTMPH0 [0:3] (0xB660; RO)

Reading this register will release the value stored in AUXSTMPH/L0 and will allow time stamping of the next value.

Table 28-169.Auxiliary Time Stamp 0 Register High—AUXSTMPH0 [0:3] (0xB660; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B660h **Offset End:** B663h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.				
07 :00	TSTH	Auxiliary Time Stamp 0 MSB value Value in 2^{32} nS resolution.			0x0	RO

28.16.1.24 Auxiliary Time Stamp 1 Register Low—AUXSTMPL1 [0:3] (0xB664; RO)

Table 28-170.Auxiliary Time Stamp 1 Register Low—AUXSTMPL1 [0:3] (0xB664; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B664h **Offset End:** B667h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TSTL	Auxiliary Time Stamp 1 LSB value Value in 1 nS resolution.			0x0	RO

28.16.1.25 Auxiliary Time Stamp 1 Register High—AUXSTMPH1 [0:3] (0xB668; RO)

Reading this register will release the value stored in AUXSTMPH/L1 and will allow stamping of the next value.

Table 28-171.Auxiliary Time Stamp 1 Register High—AUXSTMPH1 [0:3] (0xB668; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B668h **Offset End:** B66Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved Write 0 ignore on read.				
07 :00	TSTH	Auxiliary Time Stamp 1 MSB value Value in 2^{32} nS resolution.			0x0	RO

28.16.1.26 Time Sync RX Configuration—TSYNCRXCFG [0:3] (0x5F50; R/W)

Table 28-172.Time Sync RX Configuration—TSYNCRXCFG [0:3] (0x5F50; R/W)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5F50h **Offset End:** 5F53h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved				
15 :08	MSGT	V2 Message Type to timestamp			0x0	R/W
07 :00	CTRLT	V1 control to timestamp			0x0	R/W

28.16.1.27 Time Sync SDP Configuration Register—TSSDP [0:3] (0x003C; R/W)

This register defines the assignment of SDP pins to the Time sync auxiliary capabilities.

Table 28-173.Time Sync SDP Configuration Register—TSSDP [0:3] (0x003C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 003Ch **Offset End:** 003Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11	TS_SDP1_EN	When set indicates that SDP1 is assigned to Tsync.		0h	R/W
10 :09	TS_SDP1_SEL	SDP1 allocation to Tsync event – when TS_SDP1_EN is set, these bits select the Tsync event that is routed to SDP1. 00b = Target Time 0 is output on SDP1 01b = Target Time 1 is output on SDP1 10b = Freq Clock 0 is output on SDP1 11b = Freq Clock 1 is output on SDP1		0h	R/W
08	TS_SDP0_EN	When set indicates that SDP0 is assigned to Tsync.		0h	R/W
07 :06	TS_SDP0_SEL	SDP0 allocation to Tsync event – when TS_SDP0_EN is set, these bits select the Tsync event that is routed to SDP0. 00b = Target Time 0 is output on SDP0 01b = Target Time 1 is output on SDP0 10b = Freq Clock 0 is output on SDP0 11b = Freq Clock 1 is output on SDP0		0h	R/W
05	AUX1_TS_SDP_EN	When set indicates that one of the SDPs can be used as an external trigger to Aux timestamp 1 (if this bit is set to one of the SDP pins, the corresponding pin should be configured to input mode using SPD_DIR)		0h	R/W
04 :03	AUX1_SDP_SEL	Select one of the SDPs to serve as the trigger for auxiliary time stamp 1 (in AUXSTMPL1 and AUXSTMPH1 registers) 00b = SDP0 is assigned 01b = SDP1 is assigned		0h	R/W
02	AUX0_TS_SDP_EN	When set indicates that one of the SDPs can be used as an external trigger to Aux timestamp 0 (if this bit is set to one of the SDP pins, the corresponding pin should be configured to input mode using SPD_DIR)		0h	R/W
01 :00	AUX0_SDP_SEL	Select one of the SDPs to serve as the trigger for auxiliary time stamp 0 (AUXSTMPL0 and AUXSTMPH0 registers) 00b = SDP0 is assigned 01b = SDP1 is assigned		0h	R/W

28.16.2 Time Sync Interrupt Registers

28.16.2.1 Time Sync Interrupt Cause Register—TSICR [0:3] (0xB66C; RC/W1C)

Value of register is always read as 0x0. Once *ICR.Time_Sync is set,* internal value of Register should be cleared by write 1 to all bits or cleared by read to enable reception of an additional *ICR.Time_Sync* interrupt.

Table 28-174.Time Sync Interrupt Cause Register—TSICR [0:3] (0xB66C; RC/W1C)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B66Ch **Offset End:** B66Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07	TADJ	Time Adjust 0 done Set when Time Adjust to clock out 0 or 1 completed		0h	RC/W1C
06	AUTT1	Auxiliary timestamp 1 taken. Set when new timestamp is loaded into AUXSTMP 1 (auxiliary timestamp 1) register.		0h	RC/W1C
05	AUTT0	Auxiliary timestamp 0 taken. Set when new timestamp is loaded into AUXSTMP 0 (auxiliary timestamp 0) register.		0h	RC/W1C
04	TT1	Target time 1 trigger. Set when Target Time 1 (*TRGTTIML/H1*) trigger occurs.		0h	RC/W1C
03	TT0	Target time 0 trigger. Set when Target Time 0 (*TRGTTIML/H0*) trigger occurs.		0h	RC/W1C
02	RXTS	Receive Time Stamp Set when new timestamp is loaded into *RXSTMP* register		0h	RC/W1C
01	TXTS	Transmit Time Stamp Set when new timestamp is loaded into *TXSTMP* register		0h	RC/W1C
00	SYS WARP	SYSTIM Warp around. Set when SYSTIM Warp Around occurs. Warp around occurrence can be used by Software to update software time sync time.		0h	RC/W1C

28.16.2.2 Time Sync Interrupt Mask Register—TSIM [0:3] (0xB674; RW)

Table 28-175.Time Sync Interrupt Mask Register—TSIM [0:3] (0xB674; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** B674h **Offset End:** B677h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved. Write 0, ignore on read.			
07	TADJ	Time Adjust 0 done mask 0 – No Interrupt generated when *TSICR.TADJ* is set. 1 – Interrupt generated when *TSICR.TADJ* is set.		0h	RW
06	AUTT1	Auxiliary timestamp 1 taken Mask 0 – No Interrupt generated when *TSICR.AUTT1* is set. 1 – Interrupt generated when *TSICR.AUTT1* is set.		0h	RW
05	AUTT0	Auxiliary timestamp 0 taken Mask 0 – No Interrupt generated when *TSICR.AUTT0* is set. 1 – Interrupt generated when *TSICR.AUTT0* is set.		0h	RW
04	TT1	Target time 1 Trigger Mask 0 – No Interrupt generated when *TSICR.TT1* is set. 1 – Interrupt generated when *TSICR.TT1* is set.		0h	RW
03	TT0	Target time 0 Trigger Mask 0 – No Interrupt generated when *TSICR.TT0* is set. 1 – Interrupt generated when *TSICR.TT0* is set.		0h	RW
02	RXTS	Receive Time Stamp Mask 0 – No Interrupt generated when *TSICR.RXTS* is set. 1 – Interrupt generated when *TSICR.RXTS* is set.		0h	RW
01	TXTS	Reserved. Write 0, ignore on read.			
00	SYS WARP	Time Adjust 0 done mask 0 – No Interrupt generated when *TSICR.TADJ* is set. 1 – Interrupt generated when *TSICR.TADJ* is set.		0h	RW

28.16.2.3 Time Sync Interrupt Set Register—TSIS [0:3] (0xB670; WO)

TSIS register is Write Only. Writing 1 to a bit sets respective interrupt bit in *TSICR* register. Write operation causes a single interrupt event and bit is cleared internally.

Table 28-176.Time Sync Interrupt Set Register—TSIS [0:3] (0xB670; WO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** B670h **Offset End:** B673h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved. Write 0, ignore on read.			
07	TADJ	Set Time Adjust 0 done Interrupt 0 – No *TSICR.TADJ* interrupt set. 1 – *TSICR.TADJ* interrupt set.		0h	WO
06	AUTT1	Set Auxiliary Timestamp 1 Taken Interrupt 0 – No *TSICR.AUTT1* Interrupt set. 1 – *TSICR.AUTT1* Interrupt set.		0h	WO
05	AUTT0	Set Auxiliary Timestamp 0 Taken Interrupt 0 – No *TSICR.AUTT0* Interrupt set. 1 – *TSICR.AUTT0* Interrupt set.		0h	WO
04	TT1	Set Target Time 1 Trigger Interrupt 0 – No *TSICR.TT1* Interrupt set. 1 – *TSICR.TT1* Interrupt set.		0h	WO
03	TT0	Set Target Time 0 Trigger Interrupt 0 – No *TSICR.TT0* Interrupt set. 1 – *TSICR.TT0* Interrupt set.		0h	WO
02	RXTS	Set Receive Time Stamp Interrupt 0 – No *TSICR.RXTS* Interrupt set. 1 –*TSICR.RXTS* Interrupt set.		0h	WO
01	TXTS	Set Transmit Time Stamp Interrupt 0 – No *TSICR.TXTS* Interrupt set. 1 –*TSICR.TXTS* interrupt set.		0h	WO
00	SYS WARP	Set SYSTIM Warp Around Interrupt 0 – No *TSICR.SWARP* Interrupt set. 1 –*TSICR.SWARP* interrupt set.		0h	WO

28.17 PCS Registers

These registers are used to configure the SerDes, SGMII and 1000BASE-KX PCS logic. Usage of these registers is described in Section 22.5.4.1 and Section 22.5.4.3.

28.17.1 Detailed Register Descriptions

28.17.1.1 PCS Configuration—PCS_CFG [0:3] (0x4200; R/W)

Table 28-177.PCS Configuration—PCS_CFG [0:3] (0x4200; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4200h **Offset End:** 4203h	
Size: 32 bit	**Default:** 0x00000008			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	SRESET	Soft Reset Setting this bit puts all modules within the MAC in reset except the Host Interface. The Host Interface is reset via HRST. This bit is NOT self clearing; GMAC is in a reset state until this bit is cleared.		0h	R/W
30	PCS Isolate	PCS Isolate Setting this bit isolates the PCS logic from the MAC's data path. PCS control codes are still sent and received.		0h	R/W
29 :04	Reserved	Reserved			
03	PCS Enable	PCS Enable Enables the PCS logic of the MAC. Should be set in SGMII, 1000BASE-KX and SerDes mode for normal operation. Clearing this bit disables RX/TX of both data and control codes. Use this to force link down at the far end.		1h	R/W
02 :00	Reserved	Reserved			

28.17.1.2 PCS Link Control—PCS_LCTL [0:3] (0x4208; RW)

Table 28-178.PCS Link Control—PCS_LCTL [0:3] (0x4208; RW) (Sheet 1 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4208h **Offset End:** 420Bh	
Size: 32 bit	**Default:** 0x0204000C			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :26	Reserved	Reserved			
25	LINK OK FIX EN	Link OK Fix Enable Control for enabling/disabling LinkOK/SyncOK fix. Should be set for normal operation.		1h	RW
24	FAST LINK TIMER	Fast Link Timer AN timer is reduced if this bit is set.		0h	RW

Table 28-178.PCS Link Control—PCS_LCTL [0:3] (0x4208; RW) (Sheet 2 of 3)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4208h **Offset End:** 420Bh	
Size: 32 bit	**Default:** 0x0204000C			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 :21	Reserved	Reserved		0h	RW
20	AN SGMII TRIGGER	AN SGMII Trigger If this bit is cleared, then AN is not automatically triggered in SGMII mode even if SYNC fails. AN is triggered only in response to PHY messages or by a manual setting like changing the AN Enable/Restart bits.		0h	RW
19	AN SGMII BYPASS	AN SGMII Bypass If this bit is set, then IDLE detect state is bypassed during AN in SGMII mode. This reduces the acknowledge time in SGMII mode.		0h	RW
18	AN TIMEOUT EN	AN Timeout Enable This bit enables the AN Timeout feature. During AN, if the link partner does not respond with AN pages, but continues to send good IDLE symbols, then LINK UP is assumed. (This enables LINK UP condition when link partner is not AN-capable and does not affect otherwise). This bit should not be set in SGMII mode.		1h	RW
17	AN RESTART (SC)	AN Restart Used to reset/restart the link auto-negotiation process when using SerDes mode. Setting this bit restarts the Auto-negotiation process. This bit is self clearing.		0h	RW
16	AN_ENABLE	AN Enable Setting this bit enables the AN process in SerDes operating mode. Note: When link-up is forced (*CTRL.SLU*=1) the AN_ENABLE bit should be 0.		0h[1]	RW
15 :08	Reserved	Reserved		0	RW
07	Force Flow Control	Flow control mode is set according to the AN process by following Table 37-4 in the IEEE 802.3 spec. Flow control is set according to FC_TX_EN / FC_RX_EN bits in CTRL register.		0h	RW
06	LINK LATCH LOW (LL)	Link Latch Low Enable If this bit is set, then link OK going LOW (negative edge) is latched until a processor read. Afterwards, link OK is continuously updated until link OK again goes LOW (negative edge is seen).		0h	RW
05	Force Link	Force Link If this bit is set, then the internal LINK_OK variable is forced to forced link value (bit 0 of this register). Otherwise, LINK_OK is decided by internal AN/SYNC state machines.		0h	RW
04	FSD	Force Speed and Duplex If this bit is set, then speed and duplex mode is forced to forced speed value and forced duplex value, respectively. Otherwise, speed and duplex mode are decided by internal AN/SYNC state machines.		0h	RW

Table 28-178.PCS Link Control—PCS_LCTL [0:3] (0x4208; RW) (Sheet 3 of 3)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 4208h Offset End: 420Bh	
Size: 32 bit	Default: 0x0204000C				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
03	FDV	Forced Duplex Value This bit denotes the duplex mode when force speed and duplex is set. This value is also used when AN is disabled or when in SerDes mode. 1b = Full duplex (SerDes/SGMII/1000BASE-KX). 0b = Half duplex (SGMII).			1h	RW
02 :01	FSV	Forced Speed Value These bits denote the speed when force speed and duplex is set. This value is also used when AN is disabled or when in SerDes mode. 00b = 10 Mb/s (SGMII). 01b = 100 Mb/s (SGMII). 10b = 1000 Mb/s (SerDes/SGMII/1000BASE-KX). 11b = Reserved.			2h	RW
00	FLV	Forced Link Value This bit denotes the link condition when force link is set. 0b = Forced link down. 1b = Forced link up.			0h	RW

1. Read from EEPROM word 0x0F, bit 11.

28.17.1.3 PCS Link Status—PCS_LSTS [0:3] (0x420C; RO)

Table 28-179.PCS Link Status—PCS_LSTS [0:3] (0x420C; RO) (Sheet 1 of 2)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 420Ch Offset End: 420Fh	
Size: 32 bit	Default: 0x0000000C				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved				
20	AN ERROR (RWS)	AN Error This bit indicates that a AN error condition was detected in SerDes/SGMII mode. Valid after the *AN Complete* bit is set. AN error conditions: SerDes mode: Both nodes not Full Duplex SGMII mode: PHY is set to 1000 Mb/s Half Duplex mode. Software can also force a AN error condition by writing to this bit (or can clear a existing AN error condition). This bit is cleared at the start of AN.			0h	RO
19	AN REMOTE FAULT	AN Remote Fault This bit indicates that an AN page was received with a remote fault indication during an AN process. This bit cleared on reads.			0h	RO

Table 28-179.PCS Link Status—PCS_LSTS [0:3] (0x420C; RO) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 420Ch **Offset End:** 420Fh	
Size: 32 bit	**Default:** 0x0000000C			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
18	AN TIMEDOUT	AN Timed Out This bit indicates an AN process was timed out. Valid after the *AN Complete* bit is set.		0h	RO
17	AN PAGE RECEIVED	AN Page Received This bit indicates that a link partner's page was received during an AN process. This bit is cleared on reads.		0h	RO
16	AN COMPLETE	AN Complete This bit indicates that the AN process has completed.This bit is set when the AN process reached the Link OK state. It is reset upon AN restart or reset. It is set even if the AN negotiation failed and no common capabilities where found.		0h	RO
15 :05	Reserved	Reserved			
04	SYNC OK	Sync OK This bit indicates the current value of Sync OK from the PCS Sync state machine.		0h	RO
03	DUPLEX	Duplex This bit denotes the current duplex mode. 1b = Full duplex. 0b = Half duplex.		1h	RO
02 :01	SPEED	Speed This bit denotes the current operating Speed. 00b = 10 Mb/s. 01b = 100 Mb/s. 10b = 1000 Mb/s. 11b = Reserved.		2h	RO
00	LINK OK	Link OK This bit denotes the current link ok status. 0b = Link down. 1b = Link up/OK.		0h	RO

28.17.1.4 AN Advertisement—PCS_ANADV [0:3] (0x4218; R/W)

Table 28-180.AN Advertisement—PCS_ANADV [0:3] (0x4218; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4218h **Offset End:** 421Bh	
Size: 32 bit	**Default:** 0x00000020			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	NEXTP	Next Page Capable The Controller asserts this bit to request a next page transmission. The Controller clears this bit when no subsequent next pages are requested.		0h	R/W
14	Reserved	Reserved			
13 :12	RFLT	Remote Fault The Controller's remote fault condition is encoded in this field. The Controller might indicate a fault by setting a non-zero remote fault encoding and re-negotiating. 00b = No error, link OK. 01b = Link failure. 10b = Offline. 11b = Auto-negotiation error.		0h	R/W
11 :09	Reserved	Reserved			
08 :07	ASM	Local PAUSE Capabilities the Controller'sPAUSE capability is encoded in this field. 00b = No PAUSE. 01b = Symmetric PAUSE. 10b = Asymmetric PAUSE to link partner. 11b = Both symmetric and asymmetric PAUSE to the Controller.		0h[1]	R/W
06	HDCAP (RO)	Half Duplex This bit indicates that the Controller is capable of half duplex operation. This bit is tied to 0b because the Controller does not support half duplex in SerDes mode.		0h	R/W
05	FDCAP	Full Duplex Setting this bit indicates that the Controller is capable of full duplex operation. This bit should be set to 1b for normal operation.		1h	R/W
04 :00	Reserved	Reserved			

1. Loaded from EEPROM word 0x0F, bits 13:12.

28.17.1.5 Link Partner Ability—PCS_LPAB [0:3] (0x421C; RO)

Table 28-181.Link Partner Ability—PCS_LPAB [0:3] (0x421C; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 421Ch **Offset End:** 421Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	LPNEXTP	LP Next Page Capable (SerDes) The link partner asserts this bit to indicate its ability to accept next pages.		0h	RO
14	ACK	Acknowledge (SerDes) The link partner has acknowledge page reception. SGMII: Reserved.		0h	RO
13 :12	PRF	LP Remote Fault (SerDes) The link partner's remote fault condition is encoded in this field. 00b = No error, link ok. 10b = Link failure. 01b = Offline. 11b = Auto-negotiation error. SGMII [13]: Reserved		0h	RO
11 :10	SGMII SPEED	SerDes: reserved.		0h	RO
09	Reserved	Reserved			
08 :07	LPASM	LP ASMDR/LP PAUSE (SerDes) The link partner's PAUSE capability is encoded in this field. 00b = No PAUSE. 01b = Symmetric PAUSE. 10b = Asymmetric PAUSE to link partner. 11b = Both symmetric and asymmetric PAUSE to the Controller. These bits are reserved while in SGMII mode.		0h	RO
06	LPHD	LP Half Duplex (SerDes) When set to 1b, the link partner is capable of half duplex operation. When set to 0b, the link partner is not capable of half duplex mode. This bit is reserved while in SGMII mode.		0h	RO
05	LPFD	LP Full Duplex (SerDes) When set to 1b, the link partner is capable of full duplex operation. When set to 0b, the link partner is not capable of full duplex mode. This bit is reserved while in SGMII mode.		0h	RO
04 :00	Reserved	Reserved			

28.17.1.6 Next Page Transmit—PCS_NPTX [0:3] (0x4220; RW)

Table 28-182.Next Page Transmit—PCS_NPTX [0:3] (0x4220; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4220h **Offset End:** 4223h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	NXTPG	Next Page Used to indicate whether or not this is the last Next Page to be transmitted. The encoding is: 0b = Last page. 1b = Additional Next Pages follow.		0h	RW
14	Reserved	Reserved			
13	PGTYPE	Message/Unformatted Page This bit is used to differentiate a Message Page from an Unformatted Page. The encoding is: 0b = Unformatted page. 1b = Message page.		0h	RW
12	ACK2	Acknowledge 2 Used to indicate that a device has successfully received its Link Partners' Link Code Word.		0h	RW
11	TOGGLE	Toggle This bit is used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes the opposite value of the *Toggle* bit in the previously exchanged Link Code Word. The initial value of the *Toggle* bit in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word and, therefore, can assume a value of 0b or 1b. The *Toggle* bit is set as follows: 0b = Previous value of the transmitted Link Code Word when 1b 1b = Previous value of the transmitted Link Code Word when 0b.		0h	RW
10 :00	CODE	Message/Unformatted Code Field The Message Field is an 11-bit wide field that encodes 2048 possible messages. Unformatted Code Field is an 11-bit wide field that might contain an arbitrary value.		0x0	RW

28.17.1.7 Link Partner Ability Next Page—PCS_LPABNP [0:3] (0x4224; RO)

Table 28-183.Link Partner Ability Next Page—PCS_LPABNP [0:3] (0x4224; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4224h **Offset End:** 4227h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved			
15	NXTPG	Next Page Used to indicate whether or not this is the last Next Page to be transmitted. The encoding is: 0b = Last page. 1b = Additional Next Pages follow.		0	RO
14	ACK	Acknowledge The Link Partner has acknowledged Next Page reception.		0	RO
13	MSGPG	Message Page This bit is used to differentiate a Message Page from an Unformatted Page. The encoding is: 0b = Unformatted page. 1b = Message page.		0	RO
12	ACK2	Acknowledge 2 Used to indicate that a device has successfully received its Link Partners' Link Code Word.		0	RO
11	TOGGLE	Toggle This bit is used to ensure synchronization with the Link Partner during Next Page exchange. This bit always takes the opposite value of the *Toggle* bit in the previously exchanged Link Code Word. The initial value of the *Toggle* bit in the first Next Page transmitted is the inverse of bit 11 in the base Link Code Word and, therefore, can assume a value of 0b or 1b. The *Toggle* bit is set as follows: 0b = Previous value of the transmitted Link Code Word when 1b 1b = Previous value of the transmitted Link Code Word when 0b.		0	RO
10 :00	CODE	Message/Unformatted Code Field The Message Field is an 11-bit wide field that encodes 2048 possible messages. Unformatted Code Field is an 11-bit wide field that might contain an arbitrary value.		0	RO

28.17.1.8 SFP I2C Command—I2CCMD [0:3] (0x1028; R/W)

This register is used by software to read or write to the configuration registers in an SFP module when the *CTRL_EXT.I2C Enabled* bit is set to 1.

Note: According to the SFP specification, only reads are allowed from this interface; however, SFP vendors also provide a writable register through this interface (for example, PHY registers). As a result, write capability is also supported.

Table 28-184.SFP I2C Command - I2CCMD [0:3] (0x1028; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1028h **Offset End:** 102Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	E	Error This bit set is to 1b by hardware when it fails to complete an I^2C read. Reset by a software write of a command. ***Note:*** • Bit is valid only when Ready bit is set.		0h	R/W
30	Reserved	Reserved		0h	R/W
29	R	Ready Bit Set to 1b by the Controller at the end of the I^2C transaction. For example, indicates a read or write has completed. Reset by a software write of a command.		0h	R/W
28	Reset	Reset Sequence If set, sends a reset sequence before the actual read or write. This bit is self clearing. A reset sequence is defined as nine consecutive stop conditions.		0h	R/W
27	OP	Op Code 0b = I^2C write. 1b = I^2C read.		0x0	R/W
26 :24	PHYADD	Device Address bits 3:1 The actual address used is b{1010, PHYADD[2:0], X} where X = 1 for a read command and X = 0 for a write command.		0x0	R/W
23 :16	REGADD	I^2C Register Address For example, register 0, 1, 2... 255.		0x0	R/W
15 :0	DATA	Data In a write command, software places the data bits and then the MAC shifts them out to the I^2C bus. In a read command, the MAC reads these bits serially from the I^2C bus and then software reads them from this location. ***Note:*** • This field is read in byte order and not in word order.		X	R/W

28.17.1.9 SFP I2C Parameters—I2CPARAMS [0:3] (0x102C; R/W)

This register is used to set the parameters for the I^2C access to the SFP module and to allow bit banging access to the I^2C interface.

Table 28-185.SFP I2C Parameters—I2CPARAMS [0:3] (0x102C; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0102Ch **Offset End:** 0102Fh	
Size: 32 bit	**Default:** 0x00000046			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	I2C Data order	0b - *I2CCMD.DATA* field read in Byte order. 1b - *I2CCMD.DATA* field read in Word order.		0x0	R/W
30 :16	Reserved	Reserved			R/W
15	clk_stretch_dis	0b - Enable slave clock stretching support in I^2C access. 1b - Disable clock stretching support in I^2C access.		0h	R/W
14	CLK_IN (RO)	I^2C Clock In Value Reflects the value of the I2C_CLK pad. While in bit bang mode when the CLK_OE_N field is zero, this field reflects the value set in the CLK_OUT field.		X	R/W
13	CLK_OE_N	I^2C Clock Output Enable While in bit bang mode, controls the direction of the I2C_CLK pad of this port. 0b = Pad is output. 1b = Pad is input.		0h	R/W
12	DATA_IN (RO)	I^2C_DATA_IN Reflects the value of the I2C_DATA pad. While in bit bang mode when the DATA_OE_N field is zero, this field reflects the value set in the DATA_OUT field.		X	R/W
11	DATA_OE_N	I^2C_DATA_OE_N While in bit bang mode, controls the direction of the I2C_DATA pad of this port. 0b = Pad is output. 1b = Pad is input.		0h	R/W
10	DATA_OUT	I^2C_DATA While in bit bang mode and when the DATA_OE_N field is zero, controls the value driven on the I2C_DATA pad of this port.		0h	R/W
09	CLK	I^2C Clock While in bit bang mode, controls the value driven on the I2C_CLK pad of this port.		0h	R/W
08	I2CBB_EN	I^2C Bit Bang Enable If set, the I^2C_CLK and I^2C_DATA lines are controlled via the CLK, DATA and DATA_OE_N fields of this register. Otherwise, they are controlled by the hardware machine activated via the I2CCMD or MDIC registers.		0h	R/W
07 :05	Read Time	Read Time Defines the delay between a read access and the next access. The value is in microseconds. A value of Zero is not valid		2h	R/W
04 :00	Write Time	Write Time Defines the delay between a write access and the next access. The value is in milliseconds. A value of zero is not valid.		06h	R/W

28.18 Statistics Register Descriptions

All Statistics registers reset when read. In addition, they stick at 0xFFFF_FFFF when the maximum value is reached.

For the receive statistics it should be noted that a packet is indicated as received if it passes the Controller's filters and is placed into the packet buffer memory. A packet does not have to be transferred to host memory in order to be counted as received.

Due to divergent paths between interrupt-generation and logging of relevant statistics counts, it might be possible to generate an interrupt to the system for a noteworthy event prior to the associated statistics count actually being incremented. This is extremely unlikely due to expected delays associated with the system interrupt-collection and ISR delay, but might be observed as an interrupt for which statistics values do not quite make sense. Hardware guarantees that any event noteworthy of inclusion in a statistics count is reflected in the appropriate count within 1 μs; a small time-delay prior to a read of statistics might be necessary to avoid the potential for receiving an interrupt and observing an inconsistent statistics count as part of the ISR.

28.18.1 Detailed Register Descriptions

28.18.1.1 CRC Error Count—CRCERRS [0:3] (0x4000; RC)

Counts the number of receive packets with CRC errors. In order for a packet to be counted in this register, it must pass address filtering and must be 64 bytes or greater (from <Destination Address> through <CRC>, inclusively) in length. If receives are not enabled, then this register does not increment.

Table 28-186.CRC Error Count—CRCERRS [0:3] (0x4000; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04000h **Offset End:** 04003h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CEC	CRC error count		0x0	RC

28.18.1.2 Alignment Error Count—ALGNERRC [0:3] (0x4004; RC)

Counts the number of receive packets with alignment errors (the packet is not an integer number of bytes in length). In order for a packet to be counted in this register, it must pass address filtering and must be 64 bytes or greater (from <Destination Address> through <CRC>, inclusive) in length. If receives are not enabled, then this register does not increment. This register is valid only in SGMII mode during 10/100 Mb/s operation.

Table 28-187.Alignment Error Count—ALGNERRC [0:3] (0x4004; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04004h **Offset End:** 04007h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	AEC	Alignment error count			0x0	RC

28.18.1.3 Symbol Error Count—SYMERRS [0:3] (0x4008; RC)

Counts the number of symbol errors between reads. The count increases for every bad symbol received, whether or not a packet is currently being received and whether or not the link is up. When working in SerDes/SGMII/1000BASE-KX mode these statistics can be read from the SCVPC register.

Table 28-188.Symbol Error Count—SYMERRS [0:3] (0x4008; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04008h **Offset End:** 0400Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SYMERRS	Symbol Error Count			0x0	RC

28.18.1.4 Missed Packets Count—MPC [0:3] (0x4010; RC)

Counts the number of missed packets. Packets are missed when the receive FIFO has insufficient space to store the incoming packet. This can be caused because of too few buffers allocated, or because there is insufficient bandwidth on the PCI bus. Events setting this counter causes *ICR.Rx Miss*, the Receiver Overrun Interrupt, to be set. This register does not increment if receives are not enabled.

These packets are also counted in the Total Packets Received register as well as in Total Octets Received.

Table 28-189.Missed Packets Count—MPC [0:3] (0x4010; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04010h **Offset End:** 04013h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPC	Missed Packets Count		0x0	RC

28.18.1.5 Single Collision Count—SCC [0:3] (0x4014; RC)

This register counts the number of times that a successfully transmitted packet encountered a single collision. This register only increments if transmits are enabled (*TCTL.EN* is set) and the Controller is in half-duplex mode.

Table 28-190.Single Collision Count—SCC [0:3] (0x4014; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04014h **Offset End:** 04017h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SCC	Number of times a transmit encountered a single collision.		0x0	RC

28.18.1.6 Excessive Collisions Count—ECOL [0:3] (0x4018; RC)

When 16 or more collisions have occurred on a packet, this register increments, regardless of the value of collision threshold. If collision threshold is set below 16, this counter won't increment. This register only increments if transmits are enabled (*TCTL.EN* is set) and the Controller is in half-duplex mode.

Table 28-191.Excessive Collisions Count—ECOL [0:3] (0x4018; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04018h **Offset End:** 0401Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ECC	Number of packets with more than 16 collisions.		0x0	RC

28.18.1.7 Multiple Collision Count—MCC [0:3] (0x401C; RC)

This register counts the number of times that a transmit encountered more than one collision but less than 16. This register only increments if transmits are enabled (*TCTL.EN* is set) and the Controller is in half-duplex mode.

Table 28-192.Multiple Collision Count—MCC [0:3] (0x401C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0401Ch **Offset End:** 0401Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MCC	Number of times a successful transmit encountered multiple collisions.		0x0	RC

28.18.1.8 Late Collisions Count—LATECOL [0:3] (0x4020; RC)

Late collisions are collisions that occur after one slot time. This register only increments if transmits are enabled (*TCTL.EN* is set) and the Controller is in half-duplex mode.

Table 28-193.Late Collisions Count—LATECOL [0:3] (0x4020; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04020h **Offset End:** 04023h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	LCC	Number of packets with late collisions.		0x0	RC

28.18.1.9 Collision Count—COLC [0:3] (0x4028; RC)

This register counts the total number of collisions seen by the transmitter. This register only increments if transmits are enabled (*TCTL.EN* is set) and the Controller is in half-duplex mode. This register applies to clear as well as secure traffic.

Table 28-194.Collision Count—COLC [0:3] (0x4028; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04028h **Offset End:** 0402Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CCC	Total number of collisions experienced by the transmitter.		0x0	RC

28.18.1.10 Defer Count—DC [0:3] (0x4030; RC)

This register counts defer events. A defer event occurs when the transmitter cannot immediately send a packet due to the medium being busy either because another device is transmitting, the IPG timer has not expired, half-duplex deferral events, reception of XOFF frames, or the link is not up. This register only increments if transmits are enabled (*TCTL.EN* is set). This counter does not increment for streaming transmits that are deferred due to TX IPG.

Table 28-195.Defer Count—DC [0:3] (0x4030; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04030h **Offset End:** 04033h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CDC	Number of defer events.		0x0	RC

28.18.1.11 Host Transmit Discarded Packets by MAC Count—HTDPMC [0:3] (0x403C; RC)

Table 28-196.Host Transmit Discarded Packets by MAC Count—HTDPMC [0:3] (0x403C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0403Ch **Offset End:** 0403Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HTDPMC	Number of packets sent by the host but discarded by the MAC		0x0	RC

This register counts the number of packets sent by the host (and not the manageability engine) that are dropped by the MAC. This can include packets dropped because of excessive collisions or link fail events.

28.18.1.12 Receive Length Error Count—RLEC [0:3] (0x4040; RC)

This register counts receive length error events. A length error occurs if an incoming packet passes the filter criteria but is undersized or oversized. Packets less than 64 bytes are undersized. Packets over 1518/1522/1526 bytes (according to the number of VLAN tags present) are oversized if Long Packet Enable (LPE) is 0b. If LPE is 1b, then an incoming, packet is considered oversized if it exceeds the size defined in RLPML.RLPML field.

If receives are not enabled, this register does not increment. These lengths are based on bytes in the received packet from <Destination Address> through <CRC>, inclusive.

Note: Runt packets smaller than 25 bytes may not be counted by this counter.

Table 28-197.Receive Length Error Count—RLEC [0:3] (0x4040; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04040h **Offset End:** 04043h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RLEC	Number of packets with receive length errors.		0x0	RC

28.18.1.13 XON Received Count—XONRXC [0:3] (0x4048; RC)

This register counts the number of valid XON packets received. XON packets can use the global address, or the station address. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Table 28-198.XON Received Count—XONRXC [0:3] (0x4048; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04048h **Offset End:** 0404Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	XONRXC	Number of XON packets received.		0x0	RC

28.18.1.14 XON Transmitted Count—XONTXC [0:3] (0x404C; RC)

This register counts the number of XON packets transmitted. These can be either due to a full queue or due to software initiated action (using TCTL.SWXOFF). This register only increments if transmits are enabled (*TCTL.EN* is set).

Table 28-199.XON Transmitted Count—XONTXC [0:3] (0x404C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0404Ch **Offset End:** 0404Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	XONTXC	Number of XON packets transmitted.		0x0	RC

28.18.1.15 XOFF Received Count—XOFFRXC [0:3] (0x4050; RC)

This register counts the number of valid XOFF packets received. XOFF packets can use the global address or the station address. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Table 28-200.XOFF Received Count—XOFFRXC [0:3] (0x4050; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04050h **Offset End:** 04053h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	XOFFRXC	Number of XOFF packets received.		0x0	RC

28.18.1.16 XOFF Transmitted Count—XOFFTXC [0:3] (0x4054; RC)

This register counts the number of XOFF packets transmitted. These can be either due to a full queue or due to software initiated action (using TCTL.SWXOFF). This register only increments if transmits are enabled (*TCTL.EN* is set).

Table 28-201.XOFF Transmitted Count—XOFFTXC [0:3] (0x4054; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04054h **Offset End:** 04057h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	XOFFTXC	Number of XOFF packets transmitted.		0x0	RC

28.18.1.17 FC Received Unsupported Count—FCRUC [0:3] (0x4058; RC)

This register counts the number of unsupported flow control frames that are received.

The FCRUC counter increments when a flow control packet is received that matches either the reserved flow control multicast address (in FCAH/L) or the MAC station address, and has a matching flow control type field match (to the value in FCT), but has an incorrect opcode field. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Table 28-202.FC Received Unsupported Count—FCRUC [0:3] (0x4058; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04058h **Offset End:** 0405Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FCRUC	Number of unsupported flow control frames received.		0x0	RC

28.18.1.18 Packets Received [64 Bytes] Count—PRC64 [0:3] (0x405C; RC)

This register counts the number of good packets received that are exactly 64 bytes (from <Destination Address> through <CRC>, inclusive) in length. Packets that are counted in the Missed Packet Count register are not counted in this register. Packets sent to the manageability engine are included in this counter. This register does not include received flow control packets and increments only if receives are enabled (*RCTL.RXEN* is set).

Table 28-203.Packets Received [64 Bytes] Count—PRC64 [0:3] (0x405C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0405Ch **Offset End:** 0405Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC64	Number of packets received that are 64 bytes in length.		0x0	RC

28.18.1.19 Packets Received [65-127 Bytes] Count—PRC127 [0:3] (0x4060; RC)

This register counts the number of good packets received that are 65-127 bytes (from <Destination Address> through <CRC>, inclusive) in length. Packets that are counted in the Missed Packet Count register are not counted in this register. Packets sent to the manageability engine are included in this counter. This register does not include received flow control packets and increments only if receives are enabled (*RCTL.RXEN* is set).

Table 28-204.Packets Received [65-127 Bytes] Count—PRC127 [0:3] (0x4060; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04060h **Offset End:** 04063h	
Size: 32 bit	**Default:** 0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC127	Number of packets received that are 65-127 bytes in length.		0x0	RC

28.18.1.20 Packets Received [128-255 Bytes] Count—PRC255 [0:3] (0x4064; RC)

This register counts the number of good packets received that are 128-255 bytes (from <Destination Address> through <CRC>, inclusive) in length. Packets that are counted in the Missed Packet Count register are not counted in this register. Packets sent to the manageability engine are included in this counter. This register does not include received flow control packets and increments only if receives are enabled (*RCTL.RXEN* is set).

Table 28-205.Packets Received [128-255 Bytes] Count—PRC255 [0:3] (0x4064; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04064h **Offset End:** 04067h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC255	Number of packets received that are 128-255 bytes in length.		0x0	RC

28.18.1.21 Packets Received [256-511 Bytes] Count—PRC511 [0:3] (0x4068; RC)

This register counts the number of good packets received that are 256-511 bytes (from <Destination Address> through <CRC>, inclusive) in length. Packets that are counted in the Missed Packet Count register are not counted in this register. Packets sent to the manageability engine are included in this counter. This register does not include received flow control packets and increments only if receives are enabled (*RCTL.RXEN* is set).

Table 28-206.Packets Received [256-511 Bytes] Count—PRC511 [0:3] (0x4068; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04068h **Offset End:** 0406Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC511	Number of packets received that are 256-511 bytes in length.		0x0	RC

28.18.1.22 Packets Received [512-1023 Bytes] Count—PRC1023 [0:3] (0x406C; RC)

This register counts the number of good packets received that are 512-1023 bytes (from <Destination Address> through <CRC>, inclusive) in length. Packets that are counted in the Missed Packet Count register are not counted in this register. Packets sent to the manageability engine are included in this counter. This register does not include received flow control packets and increments only if receives are enabled (*RCTL.RXEN* is set).

Table 28-207.Packets Received [512-1023 Bytes] Count—PRC1023 [0:3] (0x406C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0406Ch **Offset End:** 0406Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC1023	Number of packets received that are 512-1023 bytes in length.		0x0	RC

28.18.1.23 Packets Received [1024 to Max Bytes] Count—PRC1522 [0:3] (0x4070; RC)

This register counts the number of good packets received that are from 1024 bytes to the maximum (from <Destination Address> through <CRC>, inclusive) in length. The maximum is dependent on the current receiver configuration (for example, LPE, etc.) and the type of packet being received. If a packet is counted in Receive Oversized Count, it is not counted in this register (see Section 28.18.1.35). This register does not include received flow control packets and only increments if the packet has passed address filtering and receives are enabled (*RCTL.RXEN* is set). Packets sent to the manageability engine are included in this counter.

Due to changes in the standard for maximum frame size for VLAN tagged frames in 802.3, the Controller accepts packets that have a maximum length of 1522 bytes. The RMON statistics associated with this range has been extended to count 1522 byte long packets. If CTRL.Extended_VLAN is set, packets up to 1526 bytes are counted by this counter.

Table 28-208.Packets Received [1024 to Max Bytes] Count—PRC1522 [0:3] (0x4070; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04070h **Offset End:** 04073h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PRC1522	Number of packets received that are 1024-Max bytes in length.		0x0	RC

28.18.1.24 Good Packets Received Count—GPRC [0:3] (0x4074; RC)

This register counts the number of good packets received of any legal length. The legal length for the received packet is defined by the value of Long Packet Enable (CTRL.LPE) (see Section 28.18.1.35). This register does not include received flow control packets and only counts packets that pass filtering. This register only increments if receives are enabled (*RCTL.RXEN* is set). This register does not count packets counted by the Missed Packet Count (MPC) register. Packets sent to the manageability engine are included in this counter.

Note: GPRC can count packets interrupted by a link disconnect although they have a CRC error.

Table 28-209.Good Packets Received Count—GPRC [0:3] (0x4074; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04074h **Offset End:** 04077h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPRC	Number of good packets received (of any length).		0x0	RC

28.18.1.25 Broadcast Packets Received Count—BPRC [0:3] (0x4078; RC)

This register counts the number of good (no errors) broadcast packets received. This register does not count broadcast packets received when the broadcast address filter is disabled. This register only increments if receives are enabled (*RCTL.RXEN* is set). This register does not count packets counted by the Missed Packet Count (MPC) register. Packets sent to the manageability engine are included in this counter.

Table 28-210.Broadcast Packets Received Count - BPRC [0:3] (0x4078; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04078h **Offset End:** 0407Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BPRC	Number of broadcast packets received.			0x0	RC

28.18.1.26 Multicast Packets Received Count—MPRC [0:3] (0x407C; RC)

This register counts the number of good (no errors) multicast packets received. This register does not count multicast packets received that fail to pass address filtering nor does it count received flow control packets. This register only increments if receives are enabled (*RCTL.RXEN* is set). This register does not count packets counted by the Missed Packet Count (MPC) register. Packets sent to the manageability engine are included in this counter.

Table 28-211.Multicast Packets Received Count—MPRC [0:3] (0x407C; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0407Ch **Offset End:** 0407Fh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPRC	Number of multicast packets received.			0x0	RC

28.18.1.27 Good Packets Transmitted Count—GPTC [0:3] (0x4080; RC)

This register counts the number of good (no errors) packets transmitted. A good transmit packet is considered one that is 64 or more bytes in length (from <Destination Address> through <CRC>, inclusively) in length. This does not include transmitted flow control packets. This register only increments if transmits are enabled (*TCTL.EN* is set). The register counts clear as well as secure packets.

Table 28-212.Good Packets Transmitted Count—GPTC [0:3] (0x4080; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04080h **Offset End:** 04083h	
Size: 32 bit	**Default:** 0h			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPTC	Number of good packets transmitted.		0x0	RC

28.18.1.28 Good Octets Received Count—GORCL [0:3] (0x4088; RC)

These registers make up a 64-bit register that counts the number of good (no errors) octets received. This register includes bytes received in a packet from the <Destination Address> field through the <CRC> field, inclusive; GORCL must be read before GORCH.

In addition, it sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached. Only octets of packets that pass address filtering are counted in this register. This register does not count octets of packets counted by the Missed Packet Count (MPC) register. Octets of packets sent to the manageability engine are included in this counter. This register only increments if receives are enabled (*RCTL.RXEN* is set).

These octets do not include octets of received flow control packets.

Table 28-213.Good Octets Received Count—GORCL [0:3] (0x4088; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04088h **Offset End:** 0408Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GORCL	Number of good octets received – lower 4 bytes.		0x0	RC

28.18.1.29 Good Octets Received Count—GORCH [0:3] (0x408C; RC)

Table 28-214.Good Octets Received Count—GORCH [0:3] (0x408C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0408Ch **Offset End:** 0408Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GORCH	Number of good octets received – upper 4 bytes.		0x0	RC

28.18.1.30 Good Octets Transmitted Count—GOTCL [0:3] (0x4090; RC)

These registers make up a 64-bit register that counts the number of good (no errors) packets transmitted. This register must be accessed using two independent 32-bit accesses; GOTCL must be read before GOTCH.

In addition, it sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached. This register includes bytes transmitted in a packet from the <Destination Address> field through the <CRC> field, inclusive. This register counts octets in successfully transmitted packets that are 64 or more bytes in length. This register only increments if transmits are enabled (*TCTL.EN* is set). The register counts clear as well as secure octets.

These octets do not include octets in transmitted flow control packets.

Table 28-215.Good Octets Transmitted Count—GOTCL [0:3] (0x4090; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04090h **Offset End:** 04093h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GOTCL	Number of good octets transmitted – lower 4 bytes.		0x0	RC

28.18.1.31 Good Octets Transmitted Count—GOTCH [0:3] (4094; RC)

Table 28-216.Good Octets Transmitted Count—GOTCH [0:3] (4094; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04094h **Offset End:** 04097h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GOTCH	Number of good octets transmitted – upper 4 bytes.		0x0	RC

28.18.1.32 Receive No Buffers Count—RNBC [0:3] (0x40A0; RC)

This register counts the number of times that data was not placed to host because there were no available buffers to store those frames (receive descriptor head and tail pointers were equal). The packet is still received if there is space in the FIFO. This register only increments if receives are enabled (*RCTL.RXEN* is set).

This register does not increment when flow control packets are received.

Table 28-217.Receive No Buffers Count—RNBC [0:3] (0x40A0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040A0h **Offset End:** 040A3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RNBC	Number of receive no buffer conditions		0x0	RC

28.18.1.33 Receive Undersize Count—RUC [0:3] (0x40A4; RC)

This register counts the number of received frames that passed address filtering, and were less than minimum size (64 bytes from <Destination Address> through <CRC>, inclusive), and had a valid CRC. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Table 28-218.Receive Undersize Count—RUC [0:3] (0x40A4; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040A4h **Offset End:** 040A7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RUC	Number of receive undersize errors.		0x0	RC

28.18.1.34 Receive Fragment Count—RFC [0:3] (0x40A8; RC)

This register counts the number of received frames that passed address filtering, and were less than minimum size (64 bytes from <Destination Address> through <CRC>, inclusive), but had a bad CRC (this is slightly different from the Receive Undersize Count register). This register only increments if receives are enabled (*RCTL.RXEN* is set).

Table 28-219.Receive Fragment Count—RFC [0:3] (0x40A8; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040A8h **Offset End:** 040ABh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RFC	Number of receive fragment errors.		0x0	RC

Note: Runt packets smaller than 25 bytes may not be counted by this counter.

28.18.1.35 Receive Oversize Count—ROC [0:3] (0x40AC; RC)

This register counts the number of received frames with valid CRC field that passed address filtering, and were greater than maximum size. Packets over 1522 bytes are oversized if LongPacketEnable (RCTL.LPE) is 0b. If LongPacketEnable is 1b, then an incoming packet is considered oversized if it exceeds the value set in the RLPML register.

In VMDq mode, a packet is counted only if it is bigger than the VOMLR.RLPML value for all the VMs that where supposed to receive the packet.

If receives are not enabled, this register does not increment. These lengths are based on bytes in the received packet from <Destination Address> through <CRC>, inclusive.

Note: The maximum size of a packet when LPE is 0b is fixed according to the CTRL_EXT.Extended_VLAN bit and the detection of a VLAN tag in the packet.

Table 28-220.Receive Oversize Count—ROC [0:3] (0x40AC; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040ACh **Offset End:** 040AFh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ROC	Number of receive oversize errors.		0x0	RC

28.18.1.36 Receive Jabber Count—RJC [0:3] (0x40B0; RC)

This register counts the number of received frames that passed address filtering, and were greater than maximum size and had a bad CRC (this is slightly different from the Receive Oversize Count register).

Packets over 1518/1522/1526 bytes are oversized if LPE is 0b. If LPE is 1b, then an incoming packet is considered oversized if it exceeds RLPML.LPML bytes.

If receives are not enabled, this register does not increment. These lengths are based on bytes in the received packet from <Destination Address> through <CRC>, inclusive.

Note: The maximum size of a packet when LPE is 0b is fixed according to the CTRL_EXT.Extended_VLAN bit and the detection of a VLAN tag in the packet.

Table 28-221.Receive Jabber Count—RJC [0:3] (0x40B0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040B0h **Offset End:** 040B3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RJC	Number of receive jabber errors.		0x0	RC

28.18.1.37 Management Packets Received Count—MNGPRC [0:3] (0x40B4; RC)

This register counts the total number of packets received that pass the management filters as described in the Total Cost of Ownership (TCO) System Management Bus Interface Application Note. Any packets with errors are not counted, except packets that are dropped because the management receive FIFO is full.

Packets sent to both the host and the management interface are not counted by this counter.

Table 28-222.Management Packets Received Count—MNGPRC [0:3] (0x40B4; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040B4h **Offset End:** 040B7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MNGPRC	Number of management packets received.		0x0	RC

28.18.1.38 Management Packets Dropped Count—MPDC [0:3] (0x40B8; RC)

This register counts the total number of packets received that pass the management filters as described in the Total Cost of Ownership (TCO) System Management Bus Interface Application Note, that are dropped because the management receive FIFO is full. Management packets include any packet directed to the manageability console (for example, BMC and ARP packets).

Table 28-223.Management Packets Dropped Count—MPDC [0:3] (0x40B8; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040B8h **Offset End:** 040BBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPDC	Number of management packets dropped.		0x0	RC

28.18.1.39 Management Packets Transmitted Count—MNGPTC [0:3] (0x40BC; RC)

This register counts the total number of transmitted packets originating from the manageability path.

Table 28-224.Management Packets Transmitted Count—MNGPTC [0:3] (0x40BC; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040BCh **Offset End:** 040BFh		
Size: 32 bit	**Default:** 0x0		**Power Well:** GBEAUX		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPTC	Number of management packets transmitted.		0x0	RC

28.18.1.40 Total Octets Received—TORL [0:3] (0x40C0; RC)

These registers make up a logical 64-bit register which counts the total number of octets received. This register must be accessed using two independent 32-bit accesses; TORL must be read before TORH. This register sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached.

All packets received have their octets summed into this register, regardless of their length, whether they are erred, or whether they are flow control packets. This register includes bytes received in a packet from the <Destination Address> field through the <CRC> field, inclusive. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Note: Broadcast rejected packets are counted in this counter (as opposed to all other rejected packets that are not counted).

Table 28-225.Total Octets Received—TORL [0:3] (0x40C0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040C0h **Offset End:** 040C3h		
Size: 32 bit	**Default:** 0x0		**Power Well:** GBEAUX		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TORL	Number of total octets received – lower 4 bytes.		0x0	RC

28.18.1.41 Total Octets Received—TORH [0:3] (0x40C4; RC)

Table 28-226.Total Octets Received—TORH [0:3] (0x40C4; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040C4h **Offset End:** 040C7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TORH	Number of total octets received – upper 4 bytes.		0x0	RC

28.18.1.42 Total Octets Transmitted—TOTL [0:3] (0x40C8; RC)

These registers make up a 64-bit register that counts the total number of octets transmitted. This register must be accessed using two independent 32-bit accesses; TOTL must be read before TOTH. This register sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached.

All transmitted packets have their octets summed into this register, regardless of their length or whether they are flow control packets. This register includes bytes transmitted in a packet from the <Destination Address> field through the <CRC> field, inclusive.

Octets transmitted as part of partial packet transmissions (for example, collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set).

Table 28-227.Total Octets Transmitted—TOTL [0:3] (0x40C8; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040C8h **Offset End:** 040CBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TOTL	Number of total octets transmitted – lower 4 bytes.		0x0	RC

28.18.1.43 Total Octets Transmitted—TOTH [0:3] (0x40CC; RC)

Table 28-228.Total Octets Transmitted—TOTH [0:3] (0x40CC; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040CCh **Offset End:** 040CFh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TOTH	Number of total octets transmitted – upper 4 bytes.		0x0	RC

28.18.1.44 Total Packets Received—TPR [0:3] (0x40D0; RC)

This register counts the total number of all packets received. All packets received are counted in this register, regardless of their length, whether they have errors, or whether they are flow control packets. This register only increments if receives are enabled (*RCTL.RXEN* is set).

Note: Broadcast rejected packets are counted in this counter (as opposed to all other rejected packets that are not counted).

- Runt packets smaller than 25 bytes may not be counted by this counter.

TPR can count packets interrupted by a link disconnect although they have a CRC error.

Table 28-229.Total Packets Received—TPR [0:3] (0x40D0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 040D0h **Offset End:** 040D3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TPR	Number of all packets received.		0x0	RC

28.18.1.45 Total Packets Transmitted—TPT [0:3] (0x40D4; RC)

This register counts the total number of all packets transmitted. All packets transmitted are counted in this register, regardless of their length, or whether they are flow control packets.

Partial packet transmissions (collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard packets, secure packets, packets received over the SMBus, and packets generated by the PT function.

Table 28-230.Total Packets Transmitted—TPT [0:3] (0x40D4; RC

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040D4h **Offset End:** 040D7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TPT	Number of all packets transmitted.		0x0	RC

28.18.1.46 Packets Transmitted [64 Bytes] Count—PTC64 [0:3] (0x40D8; RC)

This register counts the number of packets transmitted that are exactly 64 bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (collisions in half-duplex mode) are not included in this register. This register does not include transmitted flow control packets (which are 64 bytes in length). This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard packets, secure packets, packets received over the SMBus, and packets generated by the PT function.

Table 28-231.Packets Transmitted [64 Bytes] Count—PTC64 [0:3] (0x40D8; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040D8h **Offset End:** 040DBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC64	Number of packets transmitted that are 64 bytes in length.		0x0	RC

28.18.1.47 Packets Transmitted [65-127 Bytes] Count—PTC127 [0:3] (0x40DC; RC)

This register counts the number of packets transmitted that are 65-127 bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (for example, collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard packets, secure packets, packets received over the SMBus, and packets generated by the PT function.

Table 28-232.Packets Transmitted [65-127 Bytes] Count—PTC127 [0:3] (0x40DC; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040DCh **Offset End:** 040DFh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC127	Number of packets transmitted that are 65-127 bytes in length.		0x0	RC

28.18.1.48 Packets Transmitted [128-255 Bytes] Count—PTC255 [0:3] (0x40E0; RC)

This register counts the number of packets transmitted that are 128-255 bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard packets, secure packets, packets received over the SMBus, and packets generated by the PT function.

Table 28-233.Packets Transmitted [128-255 Bytes] Count—PTC255 [0:3] (0x40E0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040E0h **Offset End:** 040E3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC255	Number of packets transmitted that are 128-255 bytes in length.		0x0	RC

28.18.1.49 Packets Transmitted [256-511 Bytes] Count—PTC511 [0:3] (0x40E4; RC)

This register counts the number of packets transmitted that are 256-511 bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (for example, collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard and secure packets. Management packets must never be more than 200 bytes.

Table 28-234.Packets Transmitted [256-511 Bytes] Count—PTC511 [0:3] (0x40E4; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040E4h **Offset End:** 040E7h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC511	Number of packets transmitted that are 256-511 bytes in length.		0x0	RC

28.18.1.50 Packets Transmitted [512-1023 Bytes] Count—PTC1023 [0:3] (0x40E8; RC)

This register counts the number of packets transmitted that are 512-1023 bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (for example, collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard and secure packets. Management packets must never be more than 200 bytes.

Table 28-235.Packets Transmitted [512-1023 Bytes] Count—PTC1023 [0:3] (0x40E8; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040E8h **Offset End:** 040EBh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC1023	Number of packets transmitted that are 512-1023 bytes in length.		0x0	RC

28.18.1.51 Packets Transmitted [1024 Bytes or Greater] Count—PTC1522 [0:3] (0x40EC; RC)

This register counts the number of packets transmitted that are 1024 or more bytes (from <Destination Address> through <CRC>, inclusive) in length. Partial packet transmissions (for example, collisions in half-duplex mode) are not included in this register. This register only increments if transmits are enabled (*TCTL.EN* is set).

Due to changes in the standard for maximum frame size for VLAN tagged frames in 802.3, the Controller transmits packets that have a maximum length of 1522 bytes. The RMON statistics associated with this range has been extended to count 1522 byte long packets. This register counts all packets, including standard and secure packets (management packets must never be more than 200 bytes). If CTRL.Extended_VLAN is set, packets up to 1526 bytes are counted by this counter.

Table 28-236.Packets Transmitted [1024 Bytes or Greater] Count—PTC1522 [0:3] (0x40EC; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040ECh **Offset End:** 040EFh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	PTC1522	Number of packets transmitted that are 1024 or more bytes in length.		0x0	RC

28.18.1.52 Multicast Packets Transmitted Count—MPTC [0:3] (0x40F0; RC)

This register counts the number of multicast packets transmitted. This register does not include flow control packets and increments only if transmits are enabled (*TCTL.EN* is set). Counts clear as well as secure traffic.

Table 28-237.Multicast Packets Transmitted Count—MPTC [0:3] (0x40F0; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040F0h **Offset End:** 040F3h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPTC	Number of multicast packets transmitted.		0x0	RC

28.18.1.53 Broadcast Packets Transmitted Count—BPTC [0:3] (0x40F4; RC)

This register counts the number of broadcast packets transmitted. This register only increments if transmits are enabled (*TCTL.EN* is set). This register counts all packets, including standard and secure packets (management packets must never be more than 200 bytes).

Table 28-238.Broadcast Packets Transmitted Count—BPTC [0:3] (0x40F4; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040F4h **Offset End:** 040F7h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BPTC	Number of broadcast packets transmitted count.			0x0	RC

28.18.1.54 TCP Segmentation Context Transmitted Count—TSCTC [0:3] (0x40F8; RC)

This register counts the number of TCP segmentation offload transmissions and increments once the last portion of the TCP segmentation context payload is segmented and loaded as a packet into the on-chip transmit buffer. It is not a measurement of the number of packets sent out (covered by other registers). This register only increments if transmits and TCP segmentation offload are enabled.

This counter only counts pure TSO transmissions.

Table 28-239.TCP Segmentation Context Transmitted Count—TSCTC [0:3] (0x40F8; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 040F8h **Offset End:** 040FBh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	TSCTC	Number of TCP Segmentation contexts transmitted count.			0x0	RC

28.18.1.55 Interrupt Assertion Count—IAC [0:3] (0x4100; RC)

This counter counts the total number of LAN interrupts generated in the system. In case of MSI-X systems, this counter reflects the total number of MSI-X messages that are emitted.

Table 28-240.Interrupt Assertion Count—IAC [0:3] (0x4100; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04100h **Offset End:** 04103h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	IAC	This is a count of all the LAN interrupt assertions that have occurred.		0x0	RC

28.18.1.56 Rx Packets to Host Count—RPTHC [0:3] (0x4104; RC)

Table 28-241.Rx Packets to Host Count—RPTHC [0:3] (0x4104; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04104h **Offset End:** 04107h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RPTHC	This is a count of all the received packets sent to the host.		0x0	RC

28.18.1.57 Debug Counter 1—DBGC1 [0:3] (0x4108; RC)

Table 28-242.Debug Counter 1—DBGC1 [0:3] (0x4108; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04108h **Offset End:** 0410Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DBGC1	This field counts the events according to the value of the PBDIAG.STAT_SEL field. The list of possible values for this counter are described in Table 28-243:		0x0	RC

Table 28-243.DBGC1 Values

Stat Sel	Counter 1 content
0	Reserved
1	The number of Tx descriptor WB transactions performed for Q0
2	The number of Rx descriptor WB transactions performed for Q0
3	The number of Rx descriptor immediate WB transactions performed for Q0
4	The number of Tx host descriptors read by the Descriptor Handler
5	The number of Rx host descriptors read by the Descriptor Processor
6	The number of Tx data read requests done by the Dhost
7	The number of TX packets sent to DBU

28.18.1.58 Debug Counter 2—DBGC2 [0:3] (0x410C; RC)

Table 28-244.Debug Counter 2—DBGC2 [0:3] (0x410C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0410Ch **Offset End:** 0410Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DBGC2	This field counts the events according to the value of the PBDIAG.STAT_SEL field. The list of possible values for this counter are described in Table 28-245.		0x0	RC

Table 28-245.DBGC2 Values

Stat Sel	Counter 2 Content
0	Number of Rx filter packets read from DBU
1	The number of Tx descriptor WB transactions performed for Q1
2	The number of Rx descriptor WB transactions performed for Q1
3	The number of Rx descriptor immediate WB transactions performed for Q1
4	The number of Tx host descriptors written back to host
5	The number of Rx host descriptors written back to host
6	The number of Rx data write requests done by the Dhost
7	Reserved

28.18.1.59 Debug Counter 3—DBGC3 [0:3] (0x4110; RC)

Table 28-246.Debug Counter 3—DBGC3 [0:3] (0x4110; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04110h **Offset End:** 04113h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DBGC3	This field counts the events according to the value of the PBDIAG.STAT_SEL field. The list of possible values for this counter are described in Table 28-247:		0x0	RC

Table 28-247.DBGC3 Values

Stat Sel	Counter 3 Content
0	Reserved
1	The number of Tx descriptor WB transactions performed for Q2
2	The number of Rx descriptor WB transactions performed for Q2
3	The number of Rx descriptor immediate WB transactions performed for Q2
4	The number of dropped Tx packets
5	The number of Rx packets written to the DBU
6	The number of PCIe* write requests done by the DMA
7	The total amount of single send packets

28.18.1.60 Debug Counter 4—DBGC4 [0:3] (0x411C; RC)

Table 28-248.Debug Counter 4—DBGC4 [0:3] (0x411C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 0411Ch **Offset End:** 0411Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	DBGC4	This field counts the events according to the value of the PBDIAG.STAT_SEL field. The list of possible values for this counter are described in Table 28-249.		0x0	RC

Table 28-249.DBGC4 Values

Stat Sel	Counter 4 Content
0	Reserved
1	The number of Tx descriptor WB transactions performed for Q3
2	The number of Rx descriptor WB transactions performed for Q3
3	The number of Rx descriptor immediate WB transactions performed for Q3
4	The number of Tx packets read from the DBU
5	The number of Rx packets read from the DBU
6	Number of PCIe* read/write requests done by the Dhost
7	The total amount of large send packets

28.18.1.61 Host Good Packets Transmitted Count—HGPTC [0:3] (0x4118; RC)

Table 28-250.Host Good Packets Transmitted Count—HGPTC [0:3] (0x4118; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04118h **Offset End:** 0411Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HGPTC	Number of good packets transmitted by the host.		0x0	RC

This register counts the number of good (non-erred) packets transmitted sent by the host. A good transmit packet is considered one that is 64 or more bytes in length (from <Destination Address> through <CRC>, inclusively) in length. This does not include transmitted flow control packets or packets sent by the manageability engine. This register only increments if transmits are enabled (*TCTL.EN* is set).

28.18.1.62 Receive Descriptor Minimum Threshold Count—RXDMTC [0:3] (0x4120; RC)

Table 28-251.Receive Descriptor Minimum Threshold Count—RXDMTC [0:3] (0x4120; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04120h **Offset End:** 04123h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	RXDMTC	This is a count of the receive descriptor minimum threshold events		0x0	RC

This register counts the number of events where the number of descriptors in one of the Rx queues was lower than the threshold defined for this queue.

28.18.1.63 Host Good Octets Received Count—HGORCL [0:3] (0x4128; RC)

Table 28-252.Host Good Octets Received Count—HGORCL [0:3] (0x4128; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 04128h **Offset End:** 0412Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HGORCL	Number of good octets received by host – lower 4 bytes		0x0	RC

28.18.1.64 Host Good Octets Received Count—HGORCH [0:3] (0x412C; RC)

Table 28-253.Host Good Octets Received Count—HGORCH [0:3] (0x412C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 0412Ch **Offset End:** 0412Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HGORCH	Number of good octets received by host – upper 4 bytes		0x0	RC

These registers make up a logical 64-bit register which counts the number of good (non-erred) octets received. This register includes bytes received in a packet from the <Destination Address> field through the <CRC> field, inclusive. This register must be accessed using two independent 32-bit accesses.; HGORCL must be read before HGORCH.

In addition, it sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached. Only packets that pass address filtering are counted in this register. This register counts only octets of packets that reached the host. The only exception is packets dropped by the DMA because of lack of descriptors in one of the queues. These packets are included in this counter.

This register only increments if receives are enabled (*RCTL.RXEN* is set).

28.18.1.65 Host Good Octets Transmitted Count—HGOTCL [0:3] (0x4130; RC)

Table 28-254.Host Good Octets Transmitted Count—HGOTCL [0:3] (0x4130; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04130h **Offset End:** 04133h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HGOTCL	Number of good octets transmitted by host – lower 4 bytes		0x0	RC

28.18.1.66 Host Good Octets Transmitted Count - HGOTCH [0:3] (0x4134; RC)

Table 28-255.Host Good Octets Transmitted Count - HGOTCH [0:3] (0x4134; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04134h **Offset End:** 04137h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HGOTCH	Number of good octets transmitted by host – upper 4 bytes		0x0	RC

These registers make up a logical 64-bit register which counts the number of good (non-erred) packets transmitted. This register must be accessed using two independent 32-bit accesses. This register resets whenever the upper 32 bits are read (HGOTCH).

In addition, it sticks at 0xFFFF_FFFF_FFFF_FFFF when the maximum value is reached. This register includes bytes transmitted in a packet from the <Destination Address> field through the <CRC> field, inclusive. This register counts octets in successfully transmitted packets which are 64 or more bytes in length. This register only increments if transmits are enabled (*TCTL.EN* is set). The register counts clear as well as secure octets.

These octets do not include octets in transmitted flow control packets or manageability packets.

28.18.1.67 Length Error Count—LENERRS [0:3] (0x4138; RC)

Table 28-256.Length Error Count—LENERRS [0:3] (0x4138; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04138h **Offset End:** 0413Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	LENERRS	Length error count.			0x0	RC

Counts the number of receive packets with Length errors. For example, valid packets (no CRC error) with a length/Type field with a value smaller or equal to 1500 greater than the frame size. In order for a packet to be counted in this register, it must pass address filtering and must be 64 bytes or greater (from <Destination Address> through <CRC>, inclusive) in length. If receives are not enabled, then this register does not increment.

28.18.1.68 SerDes/SGMII/KX Code Violation Packet Count—SCVPC [0:3] (0x4228; RW)

This register contains the number of code violation packets received. Code violation is defined as an invalid received code in the middle of a packet.

Table 28-257.SerDes/SGMII/KX Code Violation Packet Count—SCVPC [0:3] (0x4228; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04228h **Offset End:** 0422Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	CODEVIO	Code Violation Packet Count: At any point of time this field specifies number of unknown protocol packets received. Valid only in SGMII/SerDes/1000BASE-KX modes.			0x0	RW

28.18.1.69 Switch Drop Packet Count—SDPC [0:3] (0x41A4; RC)

Table 28-258.Switch Drop Packet Count—SDPC [0:3] (0x41A4; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 041A4h **Offset End:** 041A7h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SDPC	Switch Drop Packet Count: This register counts Rx packets dropped at the pool selection stage of the switch or by the storm control mechanism. For example, packets that where not routed to any of the pools and the VT_CTL.Dis_Def_Pool is set. Valid only in VMDq mode.			0x0	RC

28.18.1.70 Virtualization Statistical Counters

The Controller supports 5 statistical counters per queue to reduce processing overhead in virtualization operating mode.

28.18.1.71 Per Queue Good Packets Received Count—VFGPRC [0:3][0:7] (0x10010 + n*0x100 [n=0...7]; RO)

This register counts the number of legal length good packets received in queue[n]. The legal length for the received packet is defined by the value of Long Packet Enable (*CTRL.LPE*) (see Section 28.18.1.35). This register does not include received flow control packets and only counts packets that pass filtering. This register only increments if receive is enabled.

Note: VFGPRC may count packets interrupted by a link disconnect although they have a CRC error.

Unlike some other statistics registers that are not allocated per VM, this register is not cleared on read. Furthermore, the register wraps around back to 0x0000 on the next increment when reaching a value of 0xFFFF and then continues normal count operation.

Table 28-259.Per Queue Good Packets Received Count—VFGPRC [0:3][0:7](0x10010 + n*0x100 [n=0...7]; RO)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 10010h at 0x100 **Offset End:** 10013h at 0x100	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPRC	Number of good packets received (of any length).			0x0	RO

28.18.1.72 Per Queue Good Packets Transmitted Count—VFGPTC [0:3][0:7] (0x10014 + n*0x100 [n=0...7]; RO)

This register counts the number of good (no errors) packets transmitted on queue[n]. A good transmit packet is considered one that is 64 or more bytes in length (from <Destination Address> through <CRC>, inclusively) in length. This does not include transmitted flow control packets. This register only increments if transmits are enabled (*TCTL.EN* is set). The register counts clear as well as secure packets. This counter includes loopback packets or packets later dropped by the MAC.

Note: Unlike some other statistic registers that are not allocated per VM, this register is not cleared on read. Furthermore, the register wraps around back to 0x0000 on the next increment when reaching a value of 0xFFFF and then continues normal count operation.

Table 28-260.Per Queue Good Packets Transmitted Count—VFGPTC [0:3][0:7](0x10014 + n*0x100 [n=0...7]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 10014h at 0x100 **Offset End:** 10017h at 0x100	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GPTC	Number of good packets transmitted.		0x0	RO

28.18.1.73 Per Queue Good Octets Received Count—VFGORC [0:3][0:7] (0x10018 + n*0x100 [n=0...7]; RO)

This register counts the number of good (no errors) octets received on queue[n]. This register includes bytes received in a packet from the <Destination Address> field through the <CRC> field, inclusive.

Only octets of packets that pass address filtering are counted in this register. This register only increments if receive is enabled.

Note: Unlike some other statistic registers that are not allocated per VM, this register is not cleared on read. Furthermore, the register wraps around back to 0x0000 on the next increment when reaching a value of 0xFFFF and then continues normal count operation.

Table 28-261.Per Queue Good Octets Received Count—VFGORC [0:3][0:7] (0x10018 + n*0x100 [n=0...7]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 10018h at 0x100 **Offset End:** 1001Bh at 0x100		
Size: 32 bit	**Default:** 0x0		**Power Well:** GBEAUX		
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GORC	Number of good octets received.		0x0	RO

28.18.1.74 Per Queue Good Octets Transmitted Count—VFGOTC [0:3][0:7] (0x10034 + n*0x100 [n=0...7]; RO)

This register counts the number of good (no errors) packets transmitted on queue[n]. This register includes bytes transmitted in a packet from the <Destination Address> field through the <CRC> field, inclusive. Register also counts any padding and VLAN tag that were added by the hardware. This register counts octets in successfully transmitted packets that are 64 or more bytes in length. Octets counted do not include octets in transmitted flow control packets. This register only increments if transmit is enabled.

Note: Unlike some other statistic registers that are not allocated per VM, this register is not cleared on read. Furthermore, the register wraps around back to 0x0000 on the next increment when reaching a value of 0xFFFF and then continues normal count operation.

Table 28-262.Per Queue Good Octets Transmitted Count—VFGOTC [0:3][0:7] (0x10034 + n*0x100 [n=0...7]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 10034h at 0x100 **Offset End:** 10037h at 0x100	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	GOTC	Number of good octets transmitted – lower 4 bytes.		0x0	RO

28.18.1.75 Per Queue Multicast Packets Received Count—VFMPRC [0:3][0:7] (0x10038 + n*0x100 [n=0...7]; RO)

This register counts the number of good (no errors) multicast packets received on queue[n]. This register does not count multicast packets received that fail to pass address filtering nor does it count received flow control packets. This register only increments if receive is enabled.

Note: Unlike some other statistic registers that are not allocated per VM, this register is not cleared on read. Furthermore, the register wraps around back to 0x0000 on the next increment when reaching a value of 0xFFFF and then continues normal count operation.

Table 28-263.Per Queue Multicast Packets Received Count—VFMPRC [0:3][0:7] (0x10038 + n*0x100 [n=0...7]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 10038h at 0x100 **Offset End:** 1003Bh at 0x100	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	MPRC	Number of multicast packets received.		0x0	RO

28.19 Manageability Statistics

This section describes a set of statistics counters used by the SMBUS interface and are not accessible to the host driver.

28.19.1 Detailed Register Descriptions

28.19.1.1 BMC Management Packets Dropped Count—BMPDC [0:3] (0x4140; RC)

This register counts the total number of packets received that pass the management filters as described in the Total Cost of Ownership (TCO) System Management Bus Interface Application Note, that are dropped because the management receive FIFO is full. Management packets include any packet directed to the manageability console (for example, BMC and ARP packets).

This register is available to the firmware only.

Table 28-264.BMC Management Packets Dropped Count—BMPDC [0:3] (0x4140; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 04140h **Offset End:** 04143h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMPDC	Number of management packets dropped.			0x0	RC

28.19.1.2 BMC Management Packets Transmitted Count—BMNGPTC [0:3] (0x4144; RC)

This register counts the total number of transmitted packets originating from the manageability path.

This register is available to the firmware only.

Table 28-265.BMC Management Packets Transmitted Count—BMNGPTC [0:3] (0x4144; RC)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4144h **Offset End:** 4147h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMNGPTC	Number of management packets transmitted.			0x0	RC

28.19.1.3 BMC Management Packets Received Count—BMNGPRC [0:3] (0x413C; RC)

This register counts the total number of packets received that pass the management filters as described in the Total Cost of Ownership (TCO) System Management Bus Interface Application Note. Any packets with errors are not counted, except packets that are dropped because the management receive FIFO is full.

This register is available to the firmware only.

Table 28-266.BMC Management Packets Received Count—BMNGPRC [0:3] (0x413C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 413Ch **Offset End:** 413Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMNGPRC	Number of management packets received.		0x0	RC

28.19.1.4 BMC Total Unicast Packets Received—BUPRC [0:3] (0x4400; RC)

This register counts the number of good (no errors) unicast packets received. This register does not count unicast packets received that fail to pass address filtering. This register only increments if receives are enabled (*RCTL.RXEN* is set). This register does not count packets counted by the Missed Packet Count (MPC) register. Packets sent to the manageability engine are included in this counter.

This register is available to the firmware only.

Table 28-267.BMC Total Unicast Packets Received—BUPRC [0:3] (0x4400; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4400h **Offset End:** 4403h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BUPRC	Number of Unicast packets received.		0x0	RC

28.19.1.5 BMC Total Multicast Packets Received—BMPRC [0:3] (0x4404; RC)

This register counts the same events as the MPRC register (Section 28.18.1.26) for the BMC usage.This register is available to the firmware only.

Table 28-268.BMC Total Multicast Packets Received—BMPRC [0:3] (0x4404; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4404h **Offset End:** 4407h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMPRC	Number of multicast packets received.		0x0	RC

28.19.1.6 BMC Total Broadcast Packets Received—BBPRC [0:3] (0x4408; RC)

This register counts the same events as the BPRC register (Section 28.18.1.25) for the BMC usage.This register is available to the firmware only.

Table 28-269.BMC Total Broadcast Packets Received—BBPRC [0:3] (0x4408; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4408h **Offset End:** 440Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BBPRC	Number of broadcast packets received.		0x0	RC

28.19.1.7 BMC Total Unicast Packets Transmitted—BUPTC [0:3] (0x440C; RC)

This register counts the number of unicast packets transmitted. This register increments only if transmits are enabled (*TCTL.EN* is set).This register is available to the firmware only

Table 28-270.BMC Total Unicast Packets Transmitted—BUPTC [0:3] (0x440C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 440Ch **Offset End:** 440Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BUPTC	Number of unicast packets transmitted.		0x0	RC

28.19.1.8 BMC Total Multicast Packets Transmitted—BMPTC [0:3] (0x4410; RC)

This register counts the same events as the MPTC register (Section 28.18.1.52) for the BMC usage.This register is available to the firmware only.

Table 28-271.BMC Total Multicast Packets Transmitted—BMPTC [0:3] (0x4410; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4410h **Offset End:** 4413h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMPTC	Number of multicast packets transmitted.		0x0	RC

28.19.1.9 BMC Total Broadcast Packets Transmitted—BBPTC [0:3] (0x4414; RC)

This register counts the same events as the BPTC register (Section 28.18.1.53) for the BMC usage.This register is available to the firmware only.

Table 28-272.BMC Total Broadcast Packets Transmitted—BBPTC [0:3] (0x4414; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4414h **Offset End:** 4417h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BBPTC	Number of broadcast packets transmitted count.		0x0	RC

28.19.1.10 BMC FCS Receive Errors—BCRCERRS [0:3] (0x4418; RC)

This register counts the same events as the CRCERRS register (Section 28.18.1.1) for the BMC usage.This register is available to the firmware only.

Table 28-273.BMC FCS Receive Errors—BCRCERRS [0:3] (0x4418; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4418h **Offset End:** 441Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BCRCERRS	CRC error count		0x0	RC

28.19.1.11 BMC Alignment Errors—BALGNERRC [0:3] (0x441C; RC)

This register counts the same events as the ALGNERRC register (Section 28.18.1.2) for the BMC usage.This register is available to the firmware only.

Table 28-274.BMC Alignment Errors—BALGNERRC [0:3] (0x441C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 441Ch **Offset End:** 441Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	AEC	Alignment error count		0x0	RC

28.19.1.12 BMC Pause XON Frames Received—BXONRXC [0:3] (0x4420; RC)

This register counts the same events as the XONRXC register (Section 28.18.1.13) for the BMC usage.This register is available to the firmware only.

Table 28-275.BMC Pause XON Frames Received—BXONRXC [0:3] (0x4420; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4420h **Offset End:** 4423h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BXONRXC	Number of XON packets received.		0x0	RC

28.19.1.13 BMC Pause XOFF Frames Received—BXOFFRXC [0:3] (0x4424; RC)

This register counts the same events as the XOFFRXC register (Section 28.18.1.15) for the BMC usage.This register is available to the firmware only.

Table 28-276.BMC Pause XOFF Frames Received—BXOFFRXC [0:3] (0x4424; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4424h **Offset End:** 4427h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BXOFFRXC	Number of XOFF packets received.		0x0	RC

28.19.1.14 BMC Pause XON Frames Transmitted—BXONTXC [0:3] (0x4428; RC)

This register counts the same events as the XONTXC register (Section 28.18.1.14) for the BMC usage.This register is available to the firmware only.

Table 28-277.BMC Pause XON Frames Transmitted—BXONTXC [0:3] (0x4428; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 4428h **Offset End:** 442Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BXONTXC	Number of XON packets transmitted.		0x0	RC

28.19.1.15 BMC Pause XOFF Frames Transmitted—BXOFFTXC [0:3] (0x442C; RC)

This register counts the same events as the XOFFTXC register (Section 28.18.1.16) for the BMC usage.This register is available to the firmware only.

Table 28-278.BMC Pause XOFF Frames Transmitted—BXOFFTXC [0:3] (0x442C; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 442Ch **Offset End:** 442Fh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BXOFFTXC	Number of XOFF packets transmitted.		0x0	RC

28.19.1.16 BMC Single Collision Transmit Frames—BSCC [0:3] (0x4430; RC)

This register counts the same events as the SCC register (Section 28.18.1.5) for the BMC usage.This register is available to the firmware only.

Table 28-279.BMC Single Collision Transmit Frames—BSCC [0:3] (0x4430; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4430h **Offset End:** 4433h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BSCC	Number of times a transmit encountered a single collision.		0x0	RC

28.19.1.17 BMC Multiple Collision Transmit Frames—BMCC [0:3] (0x4434; RC)

This register counts the same events as the MCC register (Section 28.18.1.7) for the BMC usage.This register is available to the firmware only.

Table 28-280.BMC Multiple Collision Transmit Frames—BMCC [0:3] (0x4434; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 4434h **Offset End:** 4437h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	BMCC	Number of times a successful transmit encountered multiple collisions.		0x0	RC

28.20 Wake Up Controls Registers

28.20.1 Detailed Register Descriptions

28.20.1.1 Wakeup Control Register—WUC [0:3] (0x5800; R/W)

The PME_En and PME_Status bits of this register are reset when LAN_PWR_GOOD is 0b. When AUX_PWR = 0b, this register is also reset by de-asserting PE_RST_N and during a D3 to D0 transition. The other bits are reset using the standard internal resets.

Table 28-281.Wakeup Control Register—WUC [0:3] (0x5800; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5800h **Offset End:** 5803h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved			
03	APMPME	Assert PME On APM Wakeup If set to 1b, the Controller sets the PME_Status bit in the Power Management Control / Status Register (PMCSR) and asserts PE_WAKE_N and sends a PM_PME PCIe* message when APM Wakeup is enabled (*WUC.APME* = 1) and the Controller receives a matching Magic Packet. Note: When *WUC.APMPME* is set PE_WAKE_N is asserted and a PM_PME message is sent even if *PMCSR.PME_En* is cleared.		0h[1]	R/W
02	PME_Status (R/W1C)	PME_Status This bit is set when the Controller receives a wakeup event. It is the same as the PME_Status bit in the Power Management Control / Status Register (PMCSR). Writing a 1b to this bit clears also the PME_Status bit in the PMCSR.		0h	R/W
01	PME_En	PME_En This read/write bit is used by the software device driver to enable generation of a PME event without writing to the Power Management Control / Status Register (*PMCSR*) in the PCIe* configuration space. Note: Bit reflects value of *PMCSR.PME_En* bit when the bit in the *PMCSR* register is modified. However when value of *WUC.PME_En* bit is modified by software device driver, value is not reflected in the *PMCSR.PME_En* bit.		0h	R/W
00	APME	Advance Power Management Enable If set to 1b, APM Wakeup is enabled. If this bit is set and the *APMPME* bit is cleared, reception of a magic packet asserts the *WUS.MAG* bit but does not assert a PME.		0h[1]	R/W

1. Loaded from the EEPROM.

28.20.1.2 Wakeup Filter Control Register—WUFC [0:3] (0x5808; R/W)

This register is used to enable each of the pre-defined and flexible filters for wakeup support. A value of 1b means the filter is turned on.; A value of 0b means the filter is turned off.

If the NoTCO bit is set, then any packet that passes the manageability packet filtering described in Section 27.4, "Manageability Receive Filtering", does not cause a Wake Up event even if it passes one of the Wake Up Filters. This bit is set at initialization and during any EEPROM read if the SMBus Enable bit of the EEPROM's Management Control word is 1b. Otherwise its initial value is 0b.

Table 28-282.Wakeup Filter Control Register—WUFC [0:3] (0x5808; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5808h **Offset End:** 580Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23	FLX7	Flexible Filter 7 Enable.		0h	R/W
22	FLX6	Flexible Filter 6 Enable.		0h	R/W
21	FLX5	Flexible Filter 5 Enable.		0h	R/W
20	FLX4	Flexible Filter 4 Enable.		0h	R/W
19	FLX3	Flexible Filter 3 Enable.		0h	R/W
18	FLX2	Flexible Filter 2 Enable.		0h	R/W
17	FLX1	Flexible Filter 1 Enable.		0h	R/W
16	FLX0	Flexible Filter 0 Enable.		0h	R/W
15	NoTCO	Ignore TCO/management packets for wake up.		0h	R/W
14	FLEX_HQ	Flex filters Host Queuing 0 - Do not use Flex filters for queueing decisions in D0 state. 1 - Use flex filters in queuing decisions in D0 state. Note: Should be enabled only when multi queueing is enabled (MRQC.Multiple Receive Queues = 010b or 000b).		0h	R/W
13 :08	Reserved	Reserved. Set these bits to 0b.			
07	IPv6	Directed IPv6 Packet Wakeup Enable.		0h	R/W
06	IPv4	Directed IPv4 Packet Wakeup Enable.		0h	R/W
05	ARP	ARP Request Packet Wakeup Enable.		0h	R/W
04	BC	Broadcast Wakeup Enable.		0h	R/W
03	MC	Directed Multicast Wakeup Enable.		0h	R/W
02	EX	Directed Exact Wakeup Enable.[1]		0h	R/W
01	MAG	Magic Packet Wakeup Enable.		0h	R/W
00	Reserved	Reserved			

1. If the RCTL.UPE is set, and the EX bit is set also, any unicast packet wakes up the system.

28.20.1.3 Wakeup Status Register—WUS [0:3] (0x5810; R/W1C)

This register is used to record statistics about all wakeup packets received. If a packet matches multiple criteria then multiple bits could be set. Writing a 1b to any bit clears that bit.

This register is not cleared when RST# is asserted. It is only cleared when LAN_PWR_GOOD is de-asserted or when cleared by the software device driver.

Note: If additional packets are received that matches one of the wakeup filters, after the original wakeup packet is received, the WUS register is updated with the matching filters accordingly.

Table 28-283.Wakeup Status Register—WUS [0:3] (0x5810; R/W1C)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5810h **Offset End:** 5813h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved			
23	FLX7	Flexible Filter 7 Match[1].		0h	R/W1C
22	FLX6	Flexible Filter 6 Match[1].		0h	R/W1C
21	FLX5	Flexible Filter 5 Match[1].		0h	R/W1C
20	FLX4	Flexible Filter 4 Match[1].		0h	R/W1C
19	FLX3	Flexible Filter 3 Match[1].		0h	R/W1C
18	FLX2	Flexible Filter 2 Match[1].		0h	R/W1C
17	FLX1	Flexible Filter 1 Match[1].		0h	R/W1C
16	FLX0	Flexible Filter 0 Match[1].		0h	R/W1C
15 :09	Reserved	Reserved			
08	MNG	Indicates that a manageability event that should cause a PME happened.		0h	R/W1C
07	IPv6	Directed IPv6 Packet Received.		0h	R/W1C
06	IPv4	Directed IPv4 Packet Received.		0h	R/W1C
05	ARP	ARP Request Packet Received.		0h	R/W1C
04	BC	Broadcast Packet Received.		0h	R/W1C
03	MC	Directed Multicast Packet Received The packet was a multicast packet hashed to a value that corresponded to a 1 bit in the Multicast Table Array.		0h	R/W1C
02	EX	Directed Exact Packet Received The packet's address matched one of the 16 pre-programmed exact values in the Receive Address registers.		0h	R/W1C
01	MAG	Magic Packet Received.		0h	R/W1C
00	Reserved	Reserved			

1. Bit is set only when flex filter match is detected and *WUFC.FLEX_HQ* is 0.

28.20.1.4 Wakeup Packet Length—WUPL [0:3] (0x5900; RO)

This register indicates the length of the first wakeup packet received. It is valid if one of the bits in the Wakeup Status register (WUS) is set. It is not cleared by any reset.

Table 28-284.Wakeup Packet Length—WUPL [0:3] (0x5900; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5900h **Offset End:** 5903h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved			
11 :00	LEN	Length of wakeup packet. (If jumbo frames are enabled and the packet is longer than 2047 bytes then this field is 2047.)		X	RO

28.20.1.5 Wakeup Packet Memory—WUPM [0:3][0:31] (0x5A00 + 4*n [n=0...31]; RO)

This register is read-only and it is used to store the first 128 bytes of the wakeup packet for software retrieval after system wakeup. It is not cleared by any reset.

Table 28-285.Wakeup Packet Memory—WUPM [0:3][0:31] (0x5A00 + 4*n [n=0...31]; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5A00h at 4 **Offset End:** 5A03h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	WUPD	Wakeup Packet Data		X	RO

28.20.1.6 IP Address Valid—IPAV [0:3] (0x5838; R/W)

The IP Address Valid indicates whether the IP addresses in the IP Address Table are valid.

Table 28-286.IP Address Valid—IPAV [0:3] (0x5838; R/W)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5838h **Offset End:** 583Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :17	Reserved	Reserved			
16	V60	IPv6 Address 0 Valid.		0h	R/W
15 :04	Reserved	Reserved			
03	V43	IPv4 Address 3 Valid.		0h	R/W
02	V42	IPv4 Address 2 Valid.		0h	R/W
01	V41	IPv4 Address 1 Valid.		0h	R/W
00	V40	IPv4 Address 0 Valid.		0h	R/W

28.20.1.7 IPv4 Address Table—IP4AT [0:3][0:3] (0x5840 + 8*n [n=0...3]; RW)

The IPv4 Address Table is used to store the four IPv4 addresses for the ARP/IPv4 Request packet and Directed IP packet wakeup.

Note: This table is not cleared by any reset.

Table 28-287.IPv4 Address Table—IP4AT [0:3][0:3] (0x5840 + 8*n [n=0...3]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5840h at 8 **Offset End:** 5843h at 8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	IP Address	IPv4 Address n Note: These registers are written in Big Endian order (LS byte is first on the wire and is the MS byte of the IPV4 Address).		X	R/W

28.20.1.8 IPv6 Address Table—IP6AT [0:3][0:3] (0x5880 + 4*n [n=0...3]; RW)

The IPv6 Address Table is used to store the IPv6 addresses for neighbor Discovery packet filtering and Directed IP packet wakeup.

Note: This table is not cleared by any reset.

Table 28-288.IPv6 Address Table—IP6AT [0:3][0:3] (0x5880 + 4*n [n=0...3]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5880h at 4 **Offset End:** 5883h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	IP Address	IPv6 Address bytes 4*n+1:4*n +4 Note: These registers appear in Big Endian order (LS byte, LS address is first on the wire and is the MS byte of the IPV6 Address).		0x0	R/W

Field	**Dword #**	**Address**	**Bit(s)**	**Initial Value**	**Description**
IPV6ADDR0	0	0x5880	31:0	X	IPv6 Address 0, bytes 1-4
	1	0x5884	31:0	X	IPv6 Address 0, bytes 5-8
	2	0x5888	31:0	X	IPv6 Address 0, bytes 9-12
	3	0x588C	31:0	X	IPv6 Address 0, bytes 16-13

28.20.2 Flexible Host Filter Table Registers—FHFT [0:3] (0x9000 - 0x93FC; RW)

Each of the 4 Flexible Host Filters Table registers (FHFT) contains a 128 byte pattern and a corresponding 128-bit mask array. If enabled, the first 128 bytes of the received packet are compared against the non-masked bytes in the FHFT register.

Each 128 byte filter is composed of 32 DW entries, where each 2 DWs are accompanied by an 8-bit mask, one bit per filter byte. When a bit in the 8-bit mask field is set the corresponding Byte in the filter is compared.

The 8 lsb bits of the last DW of each filter contains a length field defining the number of bytes from the beginning of the packet compared by this filter, the length field should be 8 bytes aligned value. If actual packet length is less than (length - 8) (length is the value specified by the length field), the filter fails. Otherwise, it depends on the result of actual byte comparison. The value should not be greater than 128.

Note: The length field must be 8 bytes aligned. For filtering packets shorter than 8 bytes aligned the values should be rounded up to the next 8 bytes aligned value, the hardware implementation compares 8 bytes at a time so it should get extra zero masks (if needed) until the end of the length value.

Bits 31:8 of the last DW of each filter also includes a Queueing field. When the Controller is in D0 state, the *WUFC.FLEX_HQ* bit is set to 1, *MRQC.Multiple Receive Queues* = 010b or 000b and the packet matches the flex filter, the Queueing field defines the receive queue for the packet, priority of the filter and actions to be initiated.

31 0	31 8	7 0	31 0	31 0
Reserved	Reserved	Mask [7:0]	DW 1	DW 0
Reserved	Reserved	Mask [15:8]	DW 3	DW 2
Reserved	Reserved	Mask [23:16]	DW 5	DW 4
Reserved	Reserved	Mask [31:24]	DW 7	DW 6

....

31 8	7 0	31 8	7 0	31 0	31 0
Reserved	Reserved	Reserved	Mask [127:120]	DW 29	DW 28
Queueing	Length	Reserved	Mask [127:120]	DW 31	DW 30

Accessing the FHFT registers during filter operation can result in a packet being mis-classified if the write operation collides with packet reception. It is therefore advised that the flex filters are disabled prior to changing their setup.

28.20.2.1 Flex Filter Even Data Register Fields—FEDR[0:3][0:3][0:15](0x9000 + 0x10*n[n=0..15])

Table 28-289.Flex Filter Even Data Register Fields—FEDR [0:3][0:3][0:15] (0x9000 +16*n[0..15]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 9000h at 100h at /10h **Offset End:** 9003h at 100h at /10h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FEDR	Flex Filter Even Data Register Fields		X	RW

28.20.2.2 Flex Filter Odd Data Register Fields—FODR[0:3][0:3][0:15] (0x9004 + 16*n[n=0..15])

Table 28-290.Flex Filter Odd Data Register Fields—FODR [0:3][0:3][0:15] (0x9000 +16*n[0..15];RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 9004h at 100h at /10h **Offset End:** 9007h at 100h at /10h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FODR	Flex Filter Even Data Register Fields		X	RW

28.20.2.3 Flex Filter Mask Fields—FMFR[0:3][0:3][0:15] (0x9008 + 16*n[n=0..15])

Table 28-291.Flex Filter Mask Field Register—FMFR [0:3][0:3][0:15] (0x9008 +16*n[0..15];RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 9008h at 100h at /10h **Offset End:** 900Ch at 100h at /10h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved			
07 :00	FMFR	Flex Filter Mask Field Register		X	RW

28.20.2.4 Flex Filter Queueing Field—FQFR[0:3][0:3][0:15] (0x90FC + 16*n[n=0..15])

Queueing field resides in bits 31:8 of last DW (DW 63) of flex filter. The queueing field defines the receive queue to forward the packet (*RQUEUE*), the filter priority (*FLEX_PRIO*) and additional filter actions. Operations defined in queueing field are enabled when the Controller is in D0 state, *MRQC.Multiple Receive Queues* = 010b or 000b, *WUFC.FLEX_HQ* is 1 and relevant *WUFC.FLX[n]* bit is set.

Table 28-292.Flex Filter Queueing Field—FQFR [0:3][0:3][0:15] (0x90FC + 16*n[n=0..15] ;RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 90FCh at 100h **Offset End:** 90FFh at 100h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved Write 0, ignore on read.			
24	Immediate Interrupt	Enables issuing an immediate interrupt when the Flex filter matches incoming packet.		X	RW
23 :19	Reserved	Reserved Write 0, ignore on read.			
18 :16	FLEX_PRIO	Flex filter Priority Defines the priority of the filter assuming two filters with same priority don't match. If two filters with the same priority match the incoming packet, the first filter (lowest address) is used in order to define the queue destination of this packet.		X	RW
15 :11	Reserved	Reserved Write 0, ignore on read.			
10 :08	RQUEUE	Receive Queue Defines receive queue associated with this Flex filter. When match occurs in D0 state, packet is forwarded to the receive queue.		X	RW
07 :00	Length	Length Filter Length in bytes. Should be 8 bytes aligned and not greater then 128 bytes.		X	RW

28.20.2.5 Flex Filter 0—Example

Field	Dword	Address	Bit(s)	Initial Value
Filter 0 DW0	0	0x9000	31:0	X
Filter 0 DW1	1	0x9004	31:0	X
Filter 0 Mask[7:0]	2	0x9008	7:0	X
Reserved	3	0x900C	31:0	X
Filter 0 DW2	4	0x9010	31:0	X
…				
Filter 0 DW30	60	0x90F0	31:0	X
Filter 0 DW31	61	0x90F4	31:0	X
Filter 0 Mask[127:120]	62	0x90F8	7:0	X
Length	63	0x90FC	7:0	X
Filter 0 Queueing	63	0x90FC	31:8	X

28.20.3 Flexible Host Filter Table Extended Registers—FHFT_EXT [0:3] (0x9A00 - 0x9DFC; RW)

Each of the 4 additional Flexible Host Filters Table extended registers (FHFT_EXT) contains a 128 Byte pattern and a corresponding 128-bit mask array. If enabled, the first 128 Bytes of the received packet are compared against the non-masked bytes in the FHFT_EXT register. The layout and access rules of this table are the same as in FHFT.

28.20.3.1 Flex Filter Even Data Register Extended—FHFT_EXT_FEDR (0x9A00 + 0x10*n[n=0..15])

Table 28-293.Flex Filter Even Data Register Extended—FHFT_EXT_FEDR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 9A00h at 100h at /10h **Offset End:** 9A03h at 100h at /10h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FHFT_FEDR	Flex Filter Even Data Register Fields		X	RW

28.20.3.2 Flex Filter Odd Data Register Fields—FHFT_EXT_FODR [0:3][0:3][0:15] (0x9A04 + 16*n[n=0..15])

Table 28-294. Flex Filter Odd Data Register Extended—FHFT_EXT_FODR [0:3][0:3][0:15] (0x9A00 +16*n[0..15]; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 9A04h at 100h at /10h **Offset End:** 9A07h at 100h at /10h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FHFT_FODR	Flex Filter Even Data Register Fields			X	RW

28.20.3.3 Flex Filter Mask Fields—FHFT_EXT_FMFR [0:3][0:3][0:15] (0x9A08 +16*n[0..15]; RW)

Table 28-295. Flex Filter Mask Field Extended—FHFT_EXT_FMFR [0:3][0:3][0:15] (0x9A08 +16*n[0..15]; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 9A08h at 100h at /10h **Offset End:** 9A0Ch at 100h at /10h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :08	Reserved	Reserved				
07 :00	FHFT_FMFR	Flex Filter Mask Field Register			X	RW

28.20.3.4 Flex Filter Queueing Field—FHFT_EXT_FQFR [0:3][0:3] (0x9AFC; RW)

Queueing field resides in bits 31:8 of last DW (DW 63) of flex filter. The queueing field defines the receive queue to forward the packet (*RQUEUE*), the filter priority (*FLEX_PRIO*) and additional filter actions. Operations defined in queueing field are enabled when the Controller is in D0 state, *MRQC.Multiple Receive Queues* = 010b or 000b, *WUFC.FLEX_HQ* is 1 and relevant *WUFC.FLX[n]* bit is set.

Table 28-296.Flex Filter Queueing Extended—FHFT_EXT_FQFR [0:3][0:3] (0x9AFC; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 9AFCh at 100h **Offset End:** 9AFFh at 100h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :25	Reserved	Reserved Write 0, ignore on read.			
24	Immediate Interrupt	Enables issuing an immediate interrupt when the Flex filter matches incoming packet.		X	RW
23 :19	Reserved	Reserved Write 0, ignore on read.			
18 :16	FLEX_PRIO	Flex filter Priority Defines the priority of the filter assuming two filters with same priority don't match. If two filters with the same priority match the incoming packet, the first filter (lowest address) is used in order to define the queue destination of this packet.		X	RW
15 :11	Reserved	Reserved Write 0, ignore on read.			
10 :08	RQUEUE	Receive Queue Defines receive queue associated with this Flex filter. When match occurs in D0 state, packet is forwarded to the receive queue.		X	RW
07 :00	Length	Length Filter Length in bytes. Should be 8 bytes aligned and not greater then 128 bytes.		X	RW

28.21 Management Registers

All management registers are controlled by the remote BMC for both read and write. Host accesses to the management registers are blocked for write. The attributes for the fields in this chapter refer to the BMC access rights.

28.21.1 Detailed Register Descriptions

28.21.1.1 Management VLAN TAG Value—MAVTV [0:3][0:7] (0x5010 +4*n [n=0...7]; RW)

Where "n" is the VLAN filter serial number, equal to 0,1,…7.

Table 28-297.Management VLAN TAG Value—MAVTV [0:3] [0:7] (0x5010 +4*n [n=0...7]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5010h at 4 **Offset End:** 5013h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	Reserved	Reserved Write 0, ignore on read			
11 :00	VID	Contains the VLAN ID that should be compared with the incoming packet if the corresponding bit in *MDEF* is set.		0x0	RW

The *MAVTV* registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets as described in the Management chapter.

28.21.1.2 Management Flex UDP/TCP Ports—MFUTP[0:3][0:3] (0x5030 + 4*n [n=0...3]; RW)

Where each 32-bit register (n=0,…,3) refers to two UDP/TCP port filters (register at address offset n=0 refers to UDP/TCP ports 0 and 1, register at address offset n=1 refers to UDP/TCP ports 2 and 3, etc.).

Table 28-298.Management Flex UDP/TCP Ports—MFUTP [0:3][0:3] (0x5030 + 4*n [n=0...3]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5030h at 4 **Offset End:** 5033h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	MFUTP_odd	2*n+1 Management Flex UDP/TCP port		0x0	RW
15 :00	MFUTP_even	2*n Management Flex UDP/TCP port		0x0	RW

The MFUTP registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets. See section 27.4 . Reset - The MFUTP registers are cleared on LAN_PWR_GOOD only. The initial values for this register can be loaded from the EEPROM after power-up reset.

Note: The MFUTP_even and MFUTP_odd fields should be written in network order.

28.21.1.3 Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW)

Table 28-299.Management Ethernet Type Filters—METF [0:3][0:3] (0x5060 + 4*n [n=0...3]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5060h at 4 **Offset End:** 5063h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	Reserved	Reserved			
30	Polarity	0b = Positive filter - forward packets matching this filter to the manageability block. 1b = Negative filter - block packets matching this filter from the manageability block.		0h	RW
29 :16	Reserved	Reserved			
15 :00	METF	EtherType value to be compared against the L2 EtherType field in the Rx packet.		0x0	RW

The METF registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets. See section 27.4 .

Reset - The METF registers are cleared on LAN_PWR_GOOD only. The initial values for this register might be loaded from the EEPROM after power-up reset.

28.21.1.4 Management Control Register—MANC [0:3] (0x5820; RW)

Table 28-300.Management Control Register—MANC [0:3] (0x5820; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5820h **Offset End:** 5823h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved			
26	NET_TYPE	NET TYPE: 0b = pass only un-tagged packets. 1b = pass only VLAN tagged packets. Valid only if FIXED_NET_TYPE is set.		0h	RW
25	FIXED_NET_TYPE	Fixed net type: If set, only packets matching the net type defined by the NET_TYPE field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine		0h	RW
24	EN_IPv4_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.		0h	RW
23	EN_XSUM_FILTER	Enable IPv4 address Filters – when set, the last 128 bits of the MIPAF register are used to store 4 IPv4 addresses for IPv4 filtering. When cleared, these bits store a single IPv6 filter.		0h	RW
22 :20	Reserved	Reserved. Write 0 ignore on read.			
19	RCV_ALL	Receive All Enable (promiscuous mode) When set, all the traffic received from the wire is forwarded to the manageability. This bit should be set only for debug purposes.		0h	RW
18 :17	Reserved	Reserved			
16	TCO_RESET	TCO Reset occurred. Set to 1b on a TCO Reset, to reset LAN port by BMC.		0h	RW
15	TCO_Isolate (RO)	Set to 1 on a TCO Isolate command. When the "TCO_Isolate" bit is set. Host write cycles are completed successfully on the PCIe* but silently ignored by internal logic. When FW initiates the TCO Isolate command it also initiates a FW interrupt via the ICR.MNG bit to the host and writes a value of 0x0E to the FWSM.Ext_Err_Ind field. This bit is Read Only and mirrors the value of the Isolate bit in the internal Management registers.		0h	RW
14	FW_RESET	FW Reset occurred. Set to 1b on a TCO Firmware Reset. Cleared following a BMC read operation.		0h	RW
13 :00	Reserved	Reserved			

28.21.1.5 Management Only Traffic Register—MNGONLY [0:3] (0x5864; RW)

The MNGONLY register allows exclusive filtering of certain type of traffic to the BMC. Exclusive filtering enables the BMC to define certain packets that are forwarded to the BMC but not to the host. The packets will not be forwarded to the host even if they pass the host L2 filtering process.

Each manageability decision filter (MDEF and MDEF_EXT) has a corresponding bit in the MNGONLY register. When a manageability decision filter (MDEF and MDEF_EXT) forwards a packet to manageability, it may also block the packet from being forwarded to the host if the corresponding MNGONLY bit is set.

Table 28-301.Management Only Traffic Register—MNGONLY [0:3] (0x5864; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5864h **Offset End:** 5867h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value[1]**	**Bit Access**
31 :08	Reserved	Reserved			
07 :00	Exclusive to MNG	Exclusive to MNG – when set, indicates that packets forwarded by the manageability filters to manageability are not sent to the host. Bits 0...7 correspond to decision rules defined in registers MDEF[0...7] and MDEF_EXT[0...7].		0x0	RW

1. Reset - The MNGONLY register is cleared on LAN_PWR_GOOD and firmware reset. The initial values for this register can be loaded from the EEPROM after power-up reset or firmware reset.

28.21.1.6 Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW)

Where "n" is the decision filter.

Table 28-302.Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW) (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5890h at 4 **Offset End:** 5893h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value[1]**	**Bit Access**
31	Port 0x26F	Port 0x26F - Controls the inclusion of Port 0x26F filtering in the manageability filter decision (OR section).		0h	RW
30	Port 0x298	Port 0x298 - Controls the inclusion of Port 0x298 filtering in the manageability filter decision (OR section).		0h	RW
29	Neighbor Discovery	neighbor Discovery - Controls the inclusion of neighbor Discovery filtering in the manageability filter decision (OR section). The neighbor types accepted by this filter are types 0x86, 0x87, 0x88 and 0x89.		0h	RW
28	ARP Response	ARP Response - Controls the inclusion of ARP Response filtering in the manageability filter decision (OR section).		0h	RW

Table 28-302.Manageability Decision Filters—MDEF [0:3][0:7] (0x5890 + 4*n [n=0...7]; RW) (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5890h at 4 **Offset End:** 5893h at 4	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value[1]**	**Bit Access**
27	ARP Request	ARP Request - Controls the inclusion of ARP Request filtering in the manageability filter decision (OR section).		0h	RW
26	Multicast AND	Multicast - Controls the inclusion of Multicast address filtering in the manageability filter decision (AND section). Broadcast packets are not included by this bit. The packet must pass some L2 filtering to be included by this bit – either by the MANC.MCST_PASS_L2 or by some dedicated MAC address.		0h	RW
25	Broadcast OR	Broadcast - Controls the inclusion of broadcast address filtering in the manageability filter decision (OR section).		0h	RW
24 :23	Reserved	Reserved			
22 :21	Unicast OR	Unicast - Controls the inclusion of unicast address 0 to 1 respectively in the manageability filter decision (OR section).		0x0	RW
20 :17	IPv6 Address	IPv6 Address - Controls the inclusion of IPV6 address 0 o 3 respectively in the manageability filter decision (AND section). Bit 17 corresponds to IPV6 address 0, etc Note: This field is relevant only if MANC.EN_IPv4_FILTER is cleared.		0h	RW
16 :13	IPv4 Address	IPv4 Address - Controls the inclusion of IPV4 address 0 to 3 respectively in the manageability filter decision (AND section). Bit 13 corresponds to IPV4 address 0, etc. Note: This field is relevant only if MANC.EN_IPv4_FILTER is set.		0x0	RW
12 :05	VLAN AND	VLAN - Controls the inclusion of VLAN tag 0 to 7 respectively in the manageability filter decision (AND section). Bit 5 corresponds to VLAN tag 0, etc.		0x0	RW
04	Broadcast AND	Broadcast - Controls the inclusion of broadcast address filtering in the manageability filter decision (AND section).		0h	RW
03 :02	Reserved	Reserved			
01 :00	Unicast AND	Unicast - Controls the inclusion of unicast address 0 to 1 in the manageability filter decision (AND section). Bit 0 corresponds to unicast address 0, etc.		0x0	RW

1. Default values are read from EEPROM.

28.21.1.7 Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW)

Table 28-303.Manageability Decision Filters—MDEF_EXT [0:3][0:7] (0x5930 + 4*n[n=0...7]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5930h at 4 **Offset End:** 5933h at 4	
Size: 32 bit	**Default:** 0x20000000			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**[1]	**Bit Access**
31	Reserved	Reserved			
30	Apply_to_network_traffic	This decision filter does not apply to traffic received from the network. This decision filter applies to traffic received from the network.		0h	RW
29	Flow Control Discard	0 = Apply filtering rules to packets with Flow Control EtherType. 1 = Discard packets with Flow Control EtherType. Note: Flow Control EtherType is 0x8808		1h	RW
28 :25	Reserved	Reserved			
24	Flex TCO	Flex TCO - Controls the inclusion of Flex TCO filtering in the manageability filter decision (OR section). Bit 24 corresponds to Flex TCO filter 0.		0h	RW
23 :16	Flex port	Flex port - Controls the inclusion of Flex port filtering in the manageability filter decision (OR section). Bit 16 corresponds to flex port 0, etc.		0x0	RW
15 :12	Reserved	Reserved for additional L2 EtherType OR filters.		0x0	RW
11 :08	L2 EtherType OR	L2 EtherType - Controls the inclusion of L2 EtherType filtering in the manageability filter decision (OR section).		0x0	RW
07 :04	Reserved	Reserved			
03 :00	L2 EtherType AND	L2 EtherType - Controls the inclusion of L2 EtherType filtering in the manageability filter decision (AND section).		0x0	RW

1. Default values are read from EEPROM.

28.21.1.8 Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW)

The Manageability IP Address Filter register stores IP addresses for manageability filtering. The MIPAF register can be used in two configurations, depending on the value of the *MANC. EN_IPv4_FILTER* bit:

- EN_IPv4_FILTER = 0: the last 128 bits of the register store a single IPv6 address (IPV6ADDR3)
- EN_IPv4_FILTER = 1: the last 128 bits of the register store 4 IPv4 addresses (IPV4ADDR[3:0])

MANC.EN_IPv4_FILTER = 0:

DWORD#	Address	31	0
0	0x58B0	IPV6ADDR0	
1	0x58B4		
2	0x58B8		
3	0x58BC		
4	0x58C0	IPV6ADDR1	
5	0x58C4		
6	0x58C8		
7	0x58CC		
8	0x58D0	IPV6ADDR2	
9	0x58D4		
10	0x58D8		
11	0x58DC		
12	0x58E0	IPV6ADDR3	
13	0x58E4		
14	0x58E8		
15	0x58EC		

Field definitions for 0 setting:

Field	Dword #	Address	Bit(s)	Initial Value	Description
IPV6ADDR0	0	0x58B0	31:0	X*	IPv6 Address 0, bytes 1-4 (LS byte is first on the wire)
	1	0x58B4	31:0	X*	IPv6 Address 0, bytes 5-8
	2	0x58B8	31:0	X*	IPv6 Address 0, bytes 9-12
	3	0x58BC	31:0	X*	IPv6 Address 0, bytes 13-16
IPV6ADDR1	0	0x58C0	31:0	X*	IPv6 Address 1, bytes 1-4 (LS byte is first on the wire)
	1	0x58C4	31:0	X*	IPv6 Address 1, bytes 5-8
	2	0x58C8	31:0	X*	IPv6 Address 1, bytes 9-12
	3	0x58CC	31:0	X*	IPv6 Address 1, bytes 13-16
IPV6ADDR2	0	0x58D0	31:0	X*	IPv6 Address 2, bytes 1-4 (LS byte is first on the wire)
	1	0x58D4	31:0	X*	IPv6 Address 2, bytes 5-8
	2	0x58D8	31:0	X*	IPv6 Address 2, bytes 9-12
	3	0x58DC	31:0	X*	IPv6 Address 2, bytes 13-16
IPV6ADDR3	0	0x58E0	31:0	X*	IPv6 Address 3, bytes 1-4 (LS byte is first on the wire)
	1	0x58E4	31:0	X*	IPv6 Address 3, bytes 5-8
	2	0x58E8	31:0	X*	IPv6 Address 3, bytes 9-12
	3	0x58EC	31:0	X*	IPv6 Address 3, bytes 13-16

MANC.EN_IPv4_FILTER = 1:

DWORD#	Address	31	0
0	0x58B0	IPV6ADDR0	
1	0x58B4		
2	0x58B8		
3	0x58BC		
4	0x58C0	IPV6ADDR1	
5	0x58C4		
6	0x58C8		
7	0x58CC		
8	0x58D0	IPV6ADDR2	
9	0x58D4		
10	0x58D8		
11	0x58DC		
12	0x58E0	IPV4ADDR0	
13	0x58E4	IPV4ADDR1	
14	0x58E8	IPV4ADDR2	
15	0x58EC	IPV4ADDR3	

Field definitions for 1 Setting:

Field	Dword #	Address	Bit(s)	Initial Value[1]	Description
IPV6ADDR0	0	0x58B0	31:0	X	IPv6 Address 0, bytes 1-4 (LS byte is first on the wire)
	1	0x58B4	31:0	X	IPv6 Address 0, bytes 5-8
	2	0x58B8	31:0	X	IPv6 Address 0, bytes 9-12
	3	0x58BC	31:0	X	IPv6 Address 0, bytes 16-13
IPV6ADDR1	0	0x58C0	31:0	X	IPv6 Address 1, bytes 1-4 (LS byte is first on the wire)
	1	0x58C4	31:0	X	IPv6 Address 1, bytes 5-8
	2	0x58C8	31:0	X	IPv6 Address 1, bytes 9-12
	3	0x58CC	31:0	X	IPv6 Address 1, bytes 16-13
IPV6ADDR2	0	0x58D0	31:0	X	IPv6 Address 2, bytes 1-4 (LS byte is first on the wire)
	1	0x58D4	31:0	X	IPv6 Address 2, bytes 5-8
	2	0x58D8	31:0	X	IPv6 Address 2, bytes 9-12
	3	0x58DC	31:0	X	IPv6 Address 2, bytes 16-13
IPV4ADDR0	0	0x58E0	31:0	X	IPv4 Address 0 (LS byte is first on the wire)
IPV4ADDR1	1	0x58E4	31:0	X	IPv4 Address 1 (LS byte is first on the wire)
IPV4ADDR2	2	0x58E8	31:0	X	IPv4 Address 2 (LS byte is first on the wire)
IPV4ADDR3	3	0x58EC	31:0	X	IPv4 Address 3 (LS byte is first on the wire)

1. The initial values for these registers can be loaded from the EEPROM after power-up reset. The registers are written by the BMC and not accessible to the host for writing.

Initial value:

Table 28-304.Manageability IP Address Filter—MIPAF [0:3][0:15] (0x58B0 + 4*n [n=0...15]; RW)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 58B0h at 4 Offset End: 58B3h at 4	
Size: 32 bit	Default: 0x0				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value[1]**	**Bit Access**
31 :00	IP_ADDR 4 bytes	4 bytes of IP (v6 or v4) address. i mod 4 = 0 to bytes 1 – 4 i mod 4 = 1 to bytes 5 – 8 i mod 4 = 0 to bytes 9 – 12 i mod 4 = 0 to bytes 13 – 16 where i div 4 is the index of IP address (0...3)			X	RW

1. Reset - The MNGONLY register is cleared on LAN_PWR_GOOD and firmware reset. The initial values for this register can be loaded from the EEPROM after power-up reset or firmware reset.

Reset - The registers are cleared on LAN_PWR_GOOD only.

Note: These registers should be written in network order.

28.21.1.9 Manageability MAC Address Low—MMAL [0:3] (0x5910 + 8*n [n= 0...1]; RW)

Where "n" is the exact unicast/Multicast address entry, equal to 0…1.

Table 28-305.Manageability MAC Address Low—MMAL [0:3][0:1] (0x5910 + 8*n [n= 0...1]; RW)

Description:						
View: PCI	BAR: GBEPCIBAR0[0:3]		Bus:Device:Function: B:0:1+ Index1		Offset Start: 5910h at 8 Offset End: 5913h at 8	
Size: 32 bit	Default: 0x0				Power Well: GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value[1]**	**Bit Access**
31 :00	MMAL	Manageability MAC Address Low. The lower 32 bits of the 48 bit Ethernet address.			X	RW

1. The initial values for these registers can be loaded from the EEPROM after power-up reset. The registers are written by the BMC and not accessible to the host for writing.

These registers contain the lower bits of the 48 bit Ethernet address. The MMAL registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets. See section 27.4 .

Reset - The MMAL registers are cleared on LAN_PWR_GOOD only. The initial values for this register can be loaded from the EEPROM after power-up reset.

Note: The MMAL.MMAL field should be written in network order.

28.21.1.10 Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0…1]; RW)

Where "n" is the exact unicast/Multicast address entry, equal to 0…1.

Table 28-306.Manageability MAC Address High—MMAH [0:3][0:1] (0x5914 + 8*n [n=0…1]; RW)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 5914h at 8 **Offset End:** 5917h at 8	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value[1]**	**Bit Access**
31 :16	Reserved	**Reserved**. Reads as 0. Ignored on write.		0x0	RW
15 :00	MMAH	**Manageability MAC Address High**. The upper 16 bits of the 48 bit Ethernet address.		0x0	RW

1. The initial values for these registers can be loaded from the EEPROM after power-up reset. The registers are written by the BMC and not accessible to the host for writing.

These registers contain the upper bits of the 48 bit Ethernet address. The complete address is {MMAH, MMAL}. The MMAH registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets. See section 27.4 .

Reset - The MMAL registers are cleared on LAN_PWR_GOOD only. The initial values for this register can be loaded from the EEPROM after power-up reset or firmware reset.

Note: The MMAH.MMAH field should be written in network order.

28.21.1.11 Flexible TCO Filter Table Registers—FTFT [0:3] (0x9400-0x94FC; RW)

The Flexible TCO Filter Table registers (FTFT) contains a 128 byte pattern and a corresponding 128-bit mask array. If enabled, the first 128 bytes of the received packet are compared against the non-masked bytes in the FTFT register.

The 128 byte filter is composed of 32 DW entries, where each 2 DWs are accompanied by an 8-bit mask, one bit per filter byte. The bytes in each 2 DWs are written in network order for example byte0 written to bits [7:0], byte1 to bits [15:8] etc. The mask field is set so that bit0 in the mask masks byte0, bit 1 masks byte 1 etc. A value of 1 in the mask field means that the appropriate byte in the filter should be compared to the appropriate byte in the incoming packet.

Note: The mask field must be 8 bytes aligned even if the length field is not 8 bytes aligned, as the hardware implementation compares 8 bytes at a time so it should get extra masks until the end of the next quad word. Any mask bit that is located after the length should be set to 0 indicating no comparison should be done.

- In case the actual length which is defined by the length field register and the mask bits is not 8 bytes aligned there may be a case that a packet which is shorter then the actual required length passes the flexible filter. This may occur due to comparison of up to 7 bytes that come after the packet, but are not a real part of the packet.

The last DW of the filter contains a length field defining the number of bytes from the beginning of the packet compared by this filter. If actual packet length is less than length specified by this field, the filter fails. Otherwise, it depends on the result of actual byte comparison. The value should not be greater than 128.

31 0	31 8	7 0	31 0	31 0
Reserved	Reserved	Mask [7:0]	DW 1	DW 0
Reserved	Reserved	Mask [15:8]	DW 3	DW 2
Reserved	Reserved	Mask [23:16]	DW 5	DW 4
Reserved	Reserved	Mask [31:24]	DW 7	DW 6

31 8	7 0	31 8	7 0	31 0	31 0
Reserved	Reserved	Reserved	Mask [127:120]	DW 29	DW 28
Reserved	Length	Reserved	Mask [127:120]	DW 31	DW 30

Field definitions for Filter Table Registers:

Field	Dword	Address	Bit(s)	Initial Value
Filter 0 DW0	0	0x9400	31:0	X
Filter 0 DW1	1	0x9404	31:0	X
Filter 0 Mask[7:0]	2	0x9408	7:0	X
Reserved	3	0x940C		X
Filter 0 DW2	4	0x9410	31:0	X
…				
Filter 0 DW30	60	0x94F0	31:0	X
Filter 0 DW31	61	0x94F4	31:0	X
Filter 0 Mask[127:120]	62	0x94F8	7:0	X
Length	63	0x94FC	6:0	X

The initial values for the FTFT registers can be loaded from the EEPROM after power-up reset. The FTFT registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets.

Reset - The FTFT registers are cleared on LAN_PWR_GOOD only.

28.22 Memory Error Registers Description

The Controller main internal memories are protected by error correcting code (ECC) or parity bits. The Controller contains several registers that enable and report detection of internal memory errors.

28.22.1 Parity and ECC Error Indication—PEIND [0:3] (0x1084; RC)

Table 28-307.Parity and ECC Error Indication—PEIND [0:3] (0x1084; RC)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1	**Offset Start:** 1084h **Offset End:** 1087h	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved Write 0 ignore on read.			
03	dma_parity_fatal_ind (LH)	Fatal Error detected in DMA Memory Bit is latched high and cleared on read.		0h	RC
02	pcie_parity_fatal_ind (LH)	Fatal Error detected in PCIe* Memory. Bit is latched high and cleared on read.		0h	RC
01	mng_parity_fatal_ind (LH)	Fatal Error detected in Management Memory. Bit is latched high and cleared on read.		0h	RC
00	lanport_parity_fatal_ind (LH)	Fatal Error detected in LAN port Memory. Bit is latched high and cleared on read.		0h	RC

28.22.2 Parity and ECC Indication Mask—PEINDM [0:3] (0x1088; RW)

Table 28-308.Parity and ECC Indication Mask—PEINDM [0:3] (0x1088; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 1088h **Offset End:** 108Bh	
Size: 32 bit	**Default:** 0xF				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved Write 0 ignore on read.				
03	dma_parity_fatal_ind	When set and *PEIND.dma_parity_fatal_ind* is set, enable interrupt generation by setting the *ICR.FER bit.*			1h	
02	pcie_parity_fatal_ind	When set and *PEIND.PCIe*_parity_fatal_ind* is set, enable interrupt generation by setting the *ICR.FER bit.*			1h	
01	mng_parity_fatal_ind	When set and *PEIND.mng_parity_fatal_ind* is set, enable interrupt generation by setting the *ICR.FER bit.*			1h	
00	lanport_parity_fatal_ind	When set and *PEIND.lanport_parity_fatal_ind* is set, enable interrupt generation by setting the *ICR.FER bit.*			1h	

28.22.3 DMA Transmit Descriptor Parity Status—DTPARS [0:3] (0x3510; RW1C)

Table 28-309.DMA Transmit Descriptor Parity Status—DTPARS [0:3] (0x3510; RW1C)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3510h **Offset End:** 3513h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	Reserved	Reserved. Write 0, ignore on read.				
06	par_ind_dtx_icache	dtx_icache memory parity error indication			0h	RW1C
05	par_ind_dtx_dh_rf	dtx_dh_rf memory parity error indication			0h	RW1C
04	par_ind_dtx_dpt2	dtx_dpt2 memory parity error indication			0h	RW1C
03	par_ind_dtx_temp	dtx_temp memory parity error indication			0h	RW1C
02	par_ind_dtx_hdr	dtx_hdr memory parity error indication			0h	RW1C
01	par_ind_dtx_cntxt	dtx_cntxt memory parity error indication			0h	RW1C
00	par_ind_dtx_lso	dtx_lso memory parity error indication			0h	RW1C

28.22.4 DMA Receive Descriptor Parity Status—DRPARS [0:3] (0x3514; RW1C)

Table 28-310.DMA Receive Descriptor Parity Status—DRPARS [0:3] (0x3514; RW1C)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3514h **Offset End:** 3517h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved. Write 0, ignore on read.				
03	par_ind_drx_srrctl	drx_srrctl memory parity error indication.			0h	RW1C
02	par_ind_drx_icache	drx_icache memory parity error indication.			0h	RW1C
01	par_ind_drx_dh_rf	drx_dh_rf memory parity error indication.			0h	RW1C
00	par_ind_drx_desc	drx_desc memory parity error indication.			0h	RW1C

28.22.5 Dhost Parity Status—DDPARS [0:3] (0x3518; RW1C)

Table 28-311.Dhost Parity Status—DDPARS [0:3] (0x3518; RW1C)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 3518h **Offset End:** 351Bh	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :04	Reserved	Reserved. Write 0, ignore on read.				
03	par_ind_dma_stat	dma_stat memory parity error indication.			0x0	RW1C
02	par_ind_dhost_rx_desc	dhost_rx_desc memory parity error indication.			0h	RW1C
01	par_ind_dhost_tx_data	dhost_tx_data memory parity error indication.			0h	RW1C
00	par_ind_dhost_tx_desc	dhost_tx_desc memory parity error indication.			0h	RW1C

28.22.6 Tx Packet Buffer ECC Status—TPBECCSTS [0:3] (0x345C; RW)

Table 28-312.Tx Packet Buffer ECC Status—TPBECCSTS [0:3] (0x345C; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 345Ch **Offset End:** 345Fh	
Size: 32 bit	**Default:** 0x00010000				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :27	Reserved	Reserved				
26	Pb_cor_err_sta (RC)	Status of PB Correctable Error. This bit is cleared by a read.			0h	RW
16	ECC Enable	ECC Enable for Packet Buffer.			1h	RW
15 :08	Reserved	Reserved Write 0 ignore on read				
07 :00	Corr_err_cnt (RC)	Correctable Error Count This counter is incremented every time a correctable error is detected; the counter stops after reaching 0xFF. This field is cleared by read.			0x0	RW

28.22.7 LAN Port Parity Error Control Register—LANPERRCTL [0:3] (0x5F54; RW)

Table 28-313.LAN Port Parity Error Control Register—LANPERRCTL [0:3] (0x5F54; RW)

Description:						
View: PCI	**BAR:** GBEPCIBAR0[0:3]		**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5F54h **Offset End:** 5F57h	
Size: 32 bit	**Default:** 0x0				**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**		**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved				
08 :00	mrx_flx_en	Enable mrx_flx parity error indication Note: Software should program the *FHFT, FHFT_EXT, FTFT* memories before enabling parity check, to avoid incorrect parity error detection.			0x0	RW

28.22.8 LAN Port Parity Error Status Register—LANPERRSTS [0:3] (0x5F58; RO)

Parity indication bits cleared by issuing reset.

Table 28-314.LAN Port Parity Error Status Register—LANPERRSTS [0:3] (0x5F58; RO)

Description:					
View: PCI	**BAR:** GBEPCIBAR0[0:3]	**Bus:Device:Function:** B:0:1+ Index1		**Offset Start:** 5F58h **Offset End:** 5F5Bh	
Size: 32 bit	**Default:** 0x0			**Power Well:** GBEAUX	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :15	Reserved	Reserved			
14	f_vlan	f_vlan parity error indication		0h	RO
13	f_mulc	f_mulc parity error indication		0h	RO
12	f_rss	f_rss parity error indication		0h	RO
11	f_uta	f_uta parity error indication		0h	RO
10	stat_regs	stat_regs parity error indication		0h	RO
09	retx_buf	retx_buf parity error indication		0h	RO
08 :00	mrx_flx	mrx_flx parity error indication		0x0	RO

§ §

29.0 Function 1-4 (GbE)

29.1 Introduction

This chapter outlines the GbE Controller CSRs and other topics.

The Controller is a multi-function device with the following functions:

- LAN 0
- LAN 1
- LAN 2
- LAN 3

All functions contain the following regions of the PCI configuration space:

- Mandatory PCI configuration registers
- Power management capabilities
- MSI and MSI-X capabilities
- PCIe* extended capabilities

The Controller Physical Function implements Type 0 PCIe* configuration space that is compliant with PCIe* 2.0 Specification.

Figure 29-1. GbE Controller PCI Configuration Registers (Sheet 1 of 2)

	3	2	1	0	Offset[1]
Required PCI Compatible Configuration Space	Device ID		Vendor ID		00h
	Status		Command		04h
	Class Code			Revision ID	08h
	BIST	Header Type	Master Latency Timer	Cache Line Size	0Ch
	Base Address Register 0 (GbEPCIBAR0)				10h
	Base Address Register 1 (GbEPCIBAR1)				14h
	Base Address Register 2 (GbEPCIBAR2)				18h
	Base Address Register 3 (GbEPCIBAR3)				1Ch
	Base Address Register 4 (GbEPCIBAR4)				20h
	Base Address Register 5 (GbEPCIBAR5)				24h
	Cardbus CIS Pointer				28h
	Subsystem ID		Subsystem Vendor ID		2Ch
	Expansion ROM Base Address				30h
	Reserved			Capabilities Pointer	34h
	Reserved				38h
	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch
PM Cap	Power Management Capabilities		Next Cap Pointer	PM CAP_ID	40h
	Power Management Control and Status				44h
MSI Cap	Message Control		Next Cap Pointer	MSI CAP_ID	50h
	MSI Addr				54h - 5Bh
	Reserved		MSI Data		5Ch
	Mask Bits				60h
	Pending Bits				64h
	Reserved				68h
	Reserved				6Ch
MSI-X Cap	Message Control		Next Cap Pointer	MSI-X CAP_ID	70h
	MSI-X Table Offset and BIR				74h
	MSI-X PBA Offset and BIR				78h
	IOADDR				98h
	IODATA				9Ch

Figure 29-1. GbE Controller PCI Configuration Registers (Sheet 2 of 2)

	3	2	1	0	Offset[1]
PCIe* Cap	PCI Express Capabilities Register		Next Cap Pointer	PCIe CAP_ID	A0h
	Device Capabilities				A4h
	Device Status		Device Control		A8h
	Link Capability				ACh
	Link Status		Link Control		B0h
	Slot Capabilities (Reserved)				B4h
	Slot Status (Reserved)		Slot Control (Reserved)		B8h
	Root Capabilities (Reserved)		Root Control (Reserved)		BCh
	Root Status (Reserved)				C0h
	Device Capabilities 2				C4h
	Device Status 2 (Reserved)		Device Control 2		C8h
	Link Capabilities 2 (Reserved)				CCh
	Link Status 2		Link Control 2		D0h
	Slot Capabilities 2 (Reserved)				D4h
	Slot Status 2 (Reserved)		Slot Control 2 (Reserved)		D8h
VPD	VPD Address		Next Cap Pointer	VPD CAP_ID	E0h
	VPD Data				E4h
Reserved	Reserved				E8h - FCh
AER Configuration Space	Next Capability Pointer/Capability Version		AER Capability ID		100h
	Uncorrectable Error Status				104h
	Uncorrectable Error Mask				108h
	Uncorrectable Error Severity				10Ch
	Correctable Error Status				110h
	Correctable Error Mask				114h
	Control and Capability				118h
	Header Log				11Ch
	Header Log				120h
	Header Log				124h
	Header Log				128h
Serial ID Space	Next Capability Pointer/Capability Version		Serial ID Capability ID		140h
	Serial Number Register (Lower DWORD)				144h
	Serial Number Register (Upper DWORD)				148h
ARI Configuration Space	Next Capability Pointer/Capability Version		ARI Capability ID		150
	ARI Control		ARI Capability		154h
Reserved	Reserved				158h - FFCh

Notes:
1. Any addresses not shown are Read-only 0.

A description of the registers is provided in the following sections.

29.2 Detailed Register Summary

29.2.1 PCI Views

Table 29-1. Bus B, Device D, Function 1: Summary of PCIe* GigE Configuration Registers

Offset Start	Offset End	Register ID - Description	Default Value
02h	03h	"PDID0—PF Device Identification Register (GbE0)" on page 1496	0436h

Table 29-2. Bus B, Device D, Function 2: Summary of PCIe* GigE Configuration Registers

Offset Start	Offset End	Register ID - Description	Default Value
02h	03h	"PDID1—PF Device Identification Register (GbE1)" on page 1496	0436h

Table 29-3. Bus B, Device D, Function 3: Summary of PCIe* GigE Configuration Registers

Offset Start	Offset End	Register ID - Description	Default Value
02h	03h	"PDID2—PF Device Identification Register (GbE2)" on page 1497	0436h

Table 29-4. Bus B, Device D, Function 4: Summary of PCIe* GigE Configuration Registers

Offset Start	Offset End	Register ID - Description	Default Value
02h	03h	"PDID3—PF Device Identification Register (GbE3)" on page 1497	0436h

Table 29-5. Bus B, Device D, Function 1+Index1: Summary of PCIe* GigE Configuration Registers (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"PVID[0:3]—PF Vendor Identification Register" on page 1495	8086h
04h	05h	"PPCICMD[0:3]—PF Device Command Register" on page 1498	0000h
06h	07h	"PPCISTS[0:3]—PF PCI Device Status Register" on page 1499	0010h
08h	08h	"PRID[0:3]—PF Revision ID Register" on page 1501	10h
09h	0Bh	"PCC[0:3]—PF Class Code Register" on page 1501	020000h
0Eh	0Eh	"PHDR[0:3]—PF Header Type Register" on page 1502	80h
10h	13h	"GbEPCIBAR0[0:3]—BAR0 Base Address Register" on page 1503	00000000h
14h	17h	"GbEPCIBAR1[0:3]—BAR1 Base Address Register" on page 1504	00000000h
18h	1Bh	"GbEPCIBAR2[0:3]—BAR2 Base Address Register" on page 1504	00000001h
1Ch	1Fh	"GbEPCIBAR3[0:3]—BAR3 Base Address Register" on page 1505	00000000h

Table 29-5. Bus B, Device D, Function 1+Index1: Summary of PCIe* GigE Configuration Registers (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
20h	23h	"GbEPCIBAR4[0:3]—BAR4 Base Address Register" on page 1506	00000000h
24h	27h	"GbEPCIBAR5[0:3]—BAR5 Base Address Register" on page 1507	00000000h
2Ch	2Dh	"PSVID[0:3]—PF Subsystem Vendor ID Register" on page 1507	8086h
2Eh	2Fh	"PSID[0:3]—PF Subsystem ID Register" on page 1508	0000h
34h	34h	"PCP[0:3]—PF Capabilities Pointer Register" on page 1508	40h
3Ch	3Ch	"PIRQL[0:3]—PF Interrupt Line Register" on page 1509	00h
3Dh	3Dh	"PIRQP0—PF Interrupt Pin Register 0" on page 1509	02h
3Dh	3Dh	"PIRQP1—PF Interrupt Pin Register 1" on page 1510	03h
3Dh	3Dh	"PIRQP2—PF Interrupt Pin Register 2" on page 1510	04h
3Dh	3Dh	"PIRQP3—PF Interrupt Pin Register 3" on page 1511	02h
40h	40h	"PPMCAP[0:3]—PF Power Management Capabilities ID Register" on page 1512	01h
41h	41h	"PPMCP[0:3]—PF Power Management Next Capability Pointer Register" on page 1512	50h
42h	43h	"PPMC[0:3]—PF Power Management Capabilities Register" on page 1513	C823hh
44h	47h	"PPMCSR[0:3]—PF Power Management Control and Status Register" on page 1514	00000000h
50h	50h	"PMSICID[0:3] - Message Signalled Interrupt Capability ID Register" on page 1515	05h
51h	51h	"PMSINCP[0:3]—Message Signalled Interrupt Next Capability Pointer Register" on page 1515	70h
52h	53h	"PMSICTL[0:3]—Message Signalled Interrupt Control Register" on page 1516	0180h
54h	57h	"PMSILADDR[0:3]—Message Signalled Interrupt Lower Address Register" on page 1516	00000000h
58h	5Bh	"PMSIUADDR[0:3]—Message Signalled Interrupt Upper Address Register" on page 1517	00000000h
5Ch	5Dh	"PMSIDATA[0:3]—Message Signalled Interrupt Data Register" on page 1517	0000h
60h	63h	"PMSIMSK[0:3]—Message Signalled Interrupt Mask Register" on page 1517	0000h
64h	67h	"PMSIPND[0:3]—Message Signalled Interrupt Pending Register" on page 1518	0000h
70h	70h	"PMSI-X[0:3]—PF Message Signalled Interrupt X Capability ID Register" on page 1519	11h
71h	71h	"PMSIXNCP[0:3]—PF MSIX Next Capability Pointer Register" on page 1520	A0h
72h	73h	"PMSIXCNTL[0:3]—PF Message Signalled Interrupt X Control Register" on page 1520	0009h
74h	77h	"PMSIXTBIR[0:3]—PF MSI-X Table Offset & Table BIR Register" on page 1521	00000003h
78h	7Bh	"PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register" on page 1521	00002003h
98h	9Bh	"IOADDR[0:3]—IOADDR Register" on page 1522	0h
9Ch	9Fh	"IODATA[0:3]—IODATA Register" on page 1522	0h
A0h	A0h	"PPCID[0:3]—PF PCI Express Capability Register" on page 1524	10h
A1h	A1h	"PPCP[0:3]—PF PCI Express Next Capability Pointer Register" on page 1524	00h
A2h	A3h	"PPCR[0:3]—PF PCI Express Capabilities Register" on page 1524	0002h
A4h	A7h	"PPDCAP[0:3]—PF PCI Express Device Capabilities Register" on page 1525	10008041h
A8h	A9h	"PPDCNTL[0:3]—PF PCI Express Device Control Register" on page 1527	2810h
AAh	ABh	"PPDSTAT[0:3]—PF PCI Express Device Status Register" on page 1528	0010h

Table 29-5. Bus B, Device D, Function 1+Index1: Summary of PCIe* GigE Configuration Registers (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
ACh	AFh	"PLCAPR[0:3]—PF Link Capabilities Register" on page 1529	3B502h
B0h	B1h	"PLCNTLR[0:3]—PF Link Control Register" on page 1531	0000h
B2h	B3h	"PLSR[0:3]—PF Link Status Register" on page 1532	0102h
C4h	C7h	"PDCAPR2[0:3]—PF Device Capabilities 2 Register" on page 1534	00000012h
C8h	C9h	"PDCNTR2[0:3]—PF Device Control 2 Register" on page 1535	0000h
D0h	D1h	"PLCNTLR2[0:3]—PF Link Control 2 Register" on page 1536	0000h
D2h	D3h	"PLSR2[0:3]—PF Link Status 2 Register" on page 1538	0000h
E0h	E0h	"VPDCID[0:3]—VPD Capability ID Register" on page 1539	3h
E1h	E1h	"VPDNCP[0:3]—VPD Next Capability Pointer Register" on page 1539	0h
E2h	E3h	"VPDADDR[0:3]—VPD Address Register" on page 1540	0h
E4h	E7h	"VPDDATA[0:3]—VPD Data" on page 1540	0h
100h	103h	"PPCIEAERCAPID[0:3]—PF PCI Express AER Capability ID Register" on page 1542	13810001h
104h	107h	"PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register" on page 1542	0h
108h	10Bh	"PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register" on page 1543	0h
10Ch	10Fh	"PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register" on page 1544	00062030h
110h	113h	"PPAERCS[0:3]—PF PCI Express AER Correctable Error Register" on page 1545	00h
114h	117h	"PPAERCM[0:3]—PF PCI Express AER Correctable Error Mask Register" on page 1546	2000h
118h	11Bh	"PPAERCTLCAP[0:3]—PF PCI Express AER Control and Capability Register" on page 1546	0h
11Ch	11Fh	"PPAERHDRLOG0[0:3]—PF PCI Express AER Header Log 0 Register" on page 1547	0h
120h	123h	"PPAERHDRLOG1[0:3]—PF PCI Express AER Header Log 1 Register" on page 1547	0h
124h	127h	"PPAERHDRLOG2[0:3]—PF PCI Express AER Header Log 2 Register" on page 1548	0h
128h	12Bh	"PPAERHDRLOG3[0:3]—PF PCI Express AER Header Log 3 Register" on page 1548	0h
138h	13Bh	"PARIDHDR[0:3]—PF Alternative Routing ID Capability Header" on page 1549	0001000Eh
13Ch	13Dh	"PFARICAP0—PF ARI Capabilities Register" on page 1550	0200h
13Eh	13Fh	"PARIDCTL[0:3]—PF Alternative Routing ID Control Register" on page 1553	00000000h

Table 29-6. Bus B, Device D, Function 1+Index 1: Summary of GigE Registers Mapped Through GbEPCI Memory BAR

Offset Start	Offset End	Register ID - Description	Default Value
00h	03h	"IOADDR[0:3]—IOADDR Register" on page 1522	0h
04h	07h	"IODATA[0:3]—IODATA Register" on page 1522	0h

29.3 Mandatory PCI Configuration Registers

29.3.1 Detailed Register Descriptions

29.3.1.1 PVID[0:3]—PF Vendor Identification Register

The VID Register contains the vendor identification number. This 16-bit register combined with the Device Identification Register uniquely identifies any PCI device. Writes to this register have no effect.

Table 29-7. PVID[0:3]—PF Vendor Identification Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	VID	**Vendor Identification:** This register field contains the PCI standard identification for Intel, 8086h.		8086h	RO

Note: Loaded from EEPROM offset 0x0E (Vendor ID).

29.3.1.2 PDID0—PF Device Identification Register (GbE0)

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

This is a read-only register. This field identifies individual GbE Controller functions. It has the same default value for all LAN functions and is loaded from EEPROM on power up.

Note: Loaded from EEPROM offset 0x0D.

Table 29-8. PDID0—PF Device Identification Register (GbE0)

Description:					
View: PCI 1	**BAR:** Configuration	**Bus:Device:Function:** B:D:1		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0436h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	DID15_0	**Device Identification Number:** This is a 16-bit value assigned to the Controller Device (including bit 0 defined by DID0). Bit 0 is set via fuse bit: Default DID = 0436h		0436h	RO

29.3.1.3 PDID1—PF Device Identification Register (GbE1)

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

This is a read-only register. This field identifies individual GbE Controller functions. It has the same default value for all LAN functions and is loaded from EEPROM on power up.

Note: Loaded from EEPROM offset (LAN Port 1 EEPROM section Base Address+ 0x0D).

Table 29-9. PDID1—PF Device Identification Register (GbE1)

Description:					
View: PCI 2	**BAR:** Configuration	**Bus:Device:Function:** B:D:2		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0436h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	DID15_0	**Device Identification Number:** This is a 16-bit value assigned to the Controller Device. Bit 0 is set via fuse bit: Default DID = 0436h		0436h	RO

29.3.1.4 PDID2—PF Device Identification Register (GbE2)

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

This is a read-only register. This field identifies individual GbE Controller functions. It has the same default value for all LAN functions and is loaded from EEPROM on power up. The following table describes the possible values according to the SKU and functionality of each function.

Note: Loaded from EEPROM offset (LAN Port 2EEPROM section Base Address+ 0x0D).

Table 29-10. PDID2—PF Device Identification Register (GbE2)

Description:					
View: PCI 3	**BAR:** Configuration	**Bus:Device:Function:** B:D:3		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0436h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	DID15_0	**Device Identification Number:** This is a 16-bit value assigned to the Controller Device. Bit 0 is set via fuse bit: Default DID = 0436h		0436h	RO

29.3.1.5 PDID3—PF Device Identification Register (GbE3)

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

This is a read-only register. This field identifies individual GbE Controller functions. It has the same default value for all LAN functions and is loaded from EEPROM on power up but can be written by BIOS during initialization with a different value for each port. The following table describes the possible values according to the SKU and functionality of each function.

Note: Loaded from EEPROM offset (LAN Port 3EEPROM section Base Address+ 0x0D).

Table 29-11. PDID3—PF Device Identification Register (GbE3)

Description:					
View: PCI 4	**BAR:** Configuration	**Bus:Device:Function:** B:D:4		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 0436h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	DID15_0	**Device Identification Number:** This is a 16-bit value assigned to the Controller Device. Bit 0 is set via fuse bit: Default DID = 0436h		0436h	RO

29.3.1.6 PPCICMD[0:3]—PF Device Command Register

This is a read/write register. Each function has its own command register. Unless explicitly specified, functionality is the same in all functions.

Table 29-12. PPCICMD[0:3]—PF Device Command Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 11	Reserved	Reserved		0h	RV
10	INTD	**Interrupt Disable:** Setting this bit disables generation of INTX messages by the Controller. Default value is 0 which enables the INTX message generation.		0h	RW
09	FBTB	**Fast Back-to-Back Enable:** The Controller does not implement this functionality and it is not applicable to PCIe* devices. The bit is hardwired to 0.		0h	RO
08	SER	**SERR# Enable:** When set, this bit enables the non-fatal and fatal errors detected by the Controller to be reported to the RC. The error reporting can also be enabled via the PCIe* specific bits in the PCIe* device control register (Section 29.4.6.5, "PPDCNTL[0:3]—PF PCI Express Device Control Register") The default value of this bit is 0.		0h	RW
07	Reserved	Reserved/Does not apply to PCIe*.		0h	RV
06	PER	**Parity Error Enable:** Controls the setting of the Master Data Parity Error bit in the Device Status Register (Section 29.4.6.6, "PPDSTAT[0:3]—PF PCI Express Device Status Register") The Master Data Parity Error bit is set by the EP if its Parity Error Enable bit is set and either of the following two conditions occurs: • If the EP receives a poisoned Completion from the RC • If the EP poisons a write request. If the Parity Error Enable bit is cleared, the Master Data Parity Error status bit is never set The default value of this bit is 0.		0h	RW
05	VPS	**VGA Palette Snoop Enable:** The device does not implement this functionality/Does not apply to PCIe*. The bit is hardwired to 0.		0h	RO
04	MWE	**Memory Write and Invalidate Enable:** The device does not implement this functionality/Does not apply to PCIe*. The bit is hardwired to 0.		0h	RO
03	SS	**Special Cycle Enable:** The device does not implement this functionality/Does not apply to PCIe*. The bit is hardwired to 0.		0h	RO

Table 29-12. PPCICMD[0:3]—PF Device Command Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+Index1	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
02	BM	**Bus Master Enable:** Controls the ability of the Controller to issue Memory Read/Write Requests. Clearing (0) this bit prevents the Controller from issuing any Memory Requests. Because MSIs are in-band memory writes, disabling the bus master enable bit disables MSI as well. PCIe* messages are not affected by this bit.		0h	RW
01	MEM	**Memory Space Enable**: Setting this bit enables access to the memory regions the device claims through its BARs. EP will return "unsupported request" completion status & error message in response to memory transactions it receives when this bit is clear.		0h	RW
00	IO	**I/O Space Enable:** Setting this bit enables I/O space. EP will return "unsupported request" completion status & error message in response to memory transactions it receives when this bit is clear. *Note: Bit 0 should be loaded from EEPROM - PCIe* Init Configuration 2 EEPROM Word (0x19). This register bit is RW when bit 14 of the EEPROM word is a set (default value) or in EEPROM-less mode. And it is a RO when bit 14 of the EEPROM word is cleared. The default value is always 0b.*		0h	RW

29.3.1.7 PPCISTS[0:3]—PF Device Status Register

Each function has its own status register. Unless explicitly specified, functionality is the same in all functions.

Table 29-13. PPCISTS[0:3]—PF PCI Device Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+Index1	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	**Detected Parity Error:** This bit is set by EP whenever it receives a Poisoned TLP, regardless of the state the Parity Error Enable bit in the Command register. Default value of this field is 0.		0h	RW1C
14	SSE	**Signaled System Error:** This bit is set by the EP when it sends a ERR_FATAL or ERR_NONEATAL message and the SERR bit in the Device Command register bit is set. Default value of this field is 0.		0h	RW1C
13	RMA	**Received Master Abort Status:** This bit is set when EP, as a Requestor receives a Completion with Unsupported Request Completion Status. Default value of this field is 0.		0h	RW1C

Table 29-13. PPCISTS[0:3]—PF PCI Device Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
12	RTA	**Received Target Abort Status:** This bit is set when EP, as a Requestor receives a Completion with Completer Abort Completion Status. Default value of this field is 0.		0h	RW1C
11	STA	**Signaled Target Abort Status:** This bit is set when EP completes a Request using Completer Abort Completion Status. Default value of this field is 0.		0h	RW1C
10 : 09	DST	**DEVSEL Timing:** Does not apply to PCI Express. These bits are hardwired to 0.		00b	RO
08	MDPE	**Master Data Parity Error Detected:** This bit is set by EP, as a Requestor if the Parity Error Enable bit in the Command register is 1b and either of the following two conditions occurs: • Requestor receives a Completion marked poisoned • Requestor detects a parity error in the inbound completion data fifo. • Requestor poisons a write Request If the Parity Error Enable bit is 0b, this bit is never set. Default value of this field is 0.		0h	RW1C
07	FB2B	**Fast Back-to-Back Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
06	Reserved	Reserved		0h	RV
05	MC66	**66 MHz Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
04	CL	**Capabilities List:** This bit is hardwired to 1 to indicate that the Controller has a capabilities list.		1h	RO
03	IS	**Interrupt Status:** Indicates that the EP has transmitted a INTX message and is awaiting servicing. This bit does not includes MSI's generated by the EP.		0h	RO
02 : 00	Reserved	Reserved		0h	RV

29.3.1.8 PRID[0:3]—PF Revision ID Register

The default revision ID of the Controller is 0x00. The value of the rev ID is a logic XOR between the default value and the value read from EEPROM. All LAN functions have the same revision ID.

Note: Loaded from EEPROM offset 0x1E.

Table 29-14. PRID[0:3]—PF Revision ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 08h **Offset End:** 08h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :4	RIDU	**Major Revision:** Steppings which require all masks to be regenerated. 00b: A stepping 01b: B stepping 10b: C stepping 11b: D stepping		0001b	RO
03 :0	RIDL	**Minor Revision:** Incremented for each stepping which does not modify all masks. Reset for each major revision. 00b: x0 stepping 01b: x1 stepping 10b: x2 stepping 11b: x3 stepping		0000b	RO

29.3.1.9 PCC[0:3]—PF Class Code Register

The class code is a RO hard coded value that identifies the Controller's functionality.

- LAN 0...LAN3 - 0x020000/0x010000 - Ethernet/SCSI Adapter[1]

Note: Loaded from EEPROM

Table 29-15. PCC[0:3]—PF Class Code Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 09h **Offset End:** 0Bh	
Size: 24 bit	**Default:** 020000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
23 : 0	CC	**Class Code:** 0x020000/0x010000 - Ethernet/SCSI Adaptor		020000h	RO

1. Selected according to bit 11, 12, 13 or 14 in *Device Rev ID* EEPROM word for LAN0, LAN 1, LAN2 or LAN3 respectively.

29.3.1.10 PHDR[0:3]—PF Header Type Register

This indicates if a device is single function or multifunction. EP always is shown as MF device.

Table 29-16. PHDR[0:3]—PF Header Type Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 80h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	HDR	**PCI Header Type:** The header type of the EP device. 80h = multi-function device with standard header layout.		80h	RO

29.3.2 Base Address Registers (0x10...0x27; R/W)

The Base Address registers (BARs) are used to map the Controller register space of the various functions. The Controller has a memory BAR, IO BAR, and MSI-X BAR described in the following table below. The BARs location and sizes are described in Table 29-18 and Table 29-19. The fields within each BAR are then described in Table 29-18. 32-bit addresses may be used in one register for each memory mapping window or 64-bit addresses with 2 registers for each memory mapping window depending on the *BARCTRL.BAR32* bit.

Table 29-17. GbE Controller Base Address Registers Description - LAN 0...3

Mapping Windows	Mapping Description
Memory BAR	The internal MMIO registers are accessed as direct memory mapped offsets from the Base Address register. Software can access a Dword or 64 bits.
IO BAR	All internal MMIO registers can be accessed using I/O operations. There are two 4-byte registers in the IO mapping window: Addr Reg and Data Reg accessible as Dword entities. IO BAR support depends on the *IO_Sup* bit in the EEPROM "*PCIe* Init Configuration 2*" word.
MSI-X BAR	The MSI-X vectors and Pending bit array (PBA) structures are accessed as direct memory mapped offsets from the MSI-X BAR. Software can access Dword entities.

29.3.2.1 Base Address Register Fields

All base address registers have the following fields:

Table 29-18. Base Address Registers' Fields

Field	Bits	R/W	Description
Mem / IO Space Indication	0	RO	0b = Indicates memory space 1b = Indicates I/O
Memory Type	2:1	RO	00b = 32-bit BAR written by BIOS 10b = 64-bit BAR written by BIOS

Field	Bits	R/W	Description	
Prefetch Memory	3	R	0b = Non-prefetchable space. 1b = Prefetchable space. The Controller implements Non-prefetchable space in 32-bit memory BAR mode, since it has read-side effects. This bit is written by BIOS. It is required to be set to 1 for 64-bit memory BARs.	
Address Space (Low register for 64bit Memory BARs)	31:4	R/W	The length of the RW bits and RO 0b bits depend on the mapping window sizes. Init value of the RW fields is 0x0.	
			Mapping Window	RO bits
			Memory CSR	16:4 for 128KB Other values are reserved
			MSI-X space is 16KB	13:4
			I/O spaces size is 32 bytes	4:0

29.3.2.2 GbEPCIBAR0[0:3]—BAR0 Base Address Register

See Table 29-18 and Table 29-19. This register is used as either a 32-bit BAR or the lower 32-bit of a 64-bit BAR. The Prefetchable and Addressing Type fields are dependent on the BAR width.

Table 29-19. GbEPCIBAR0[0:3]—BAR0 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 10h **Offset End:** 13h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 17	ADDR	**Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
16 : 04	ZERO	**Lower Bits: Hardwired to 0 (128KB region).**		0h	RO
03	PREF	**Prefetchable:** 0: non-prefetchable (default for 32-bit) 1: prefetchable (default for 64-bit) ***Note:*** BARCTRL.BAR32 controls reset value. When BARCTRL.BAR32=1 then PREF=0. When BARCTRL.BAR32=0 then PREF=BARCTRL.PREFBAR.		0b	RO
02 : 01	TYP	**Addressing Type:** 00 - 32-bit (default for 32-bit) 10 - 64-bit (default for 64-bit) ***Note:*** BARCTRL.BAR32 controls reset value.		00b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

29.3.2.3 GbEPCIBAR1[0:3]—BAR1 Base Address Register

See Table 29-18 and Table 29-19. This register is reserved when BAR0 is setup as 32-bit. It is has RW attributes when BAR0 is setup as 64-bit. BARCTRL.BAR32 is used as the qualifier signal.

Table 29-20. GbEPCIBAR1[0:3]—BAR1 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 14h **Offset End:** 17h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 00	ADDRRES	Reserved for 32-bit (RV for 32-bit BARs) Upper Address for 64-bit (RW for 64-bit BARs) ***Note:*** BARCTRL.BAR32 controls these bits attributes.		0h	RV

29.3.2.4 GbEPCIBAR2[0:3]—BAR2 Base Address Register

See Table 29-18 and Table 29-19. This is always fixed as an IO BAR.

Table 29-21. GbEPCIBAR2[0:3]—BAR2 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :07	ADDR	**Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
06 :02	ZERO	Lower Bits: Hardwired to 0 (128 Bytes)		000b	RO
01	RES	Reserved		0b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 1 for I/O region		1b	RO

29.3.2.5 GbEPCIBAR3[0:3]—BAR3 Base Address Register

See Table 29-18 and Table 29-19. This register is used as either a 32-bit BAR or is reserved. For 64-bit BAR support BAR4-5 are used for this function.

Table 29-22. GbEPCIBAR3[0:3]—BAR3 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 14	ADDR	**Programmable Base Address:** These bits are set by BIOS to locate the base address of the region. RW - for 32-bit BARs. ***Note:*** BARCTRL.BAR32 controls these bits attributes.		0h	RW
13 : 04	RES	**Hardwired to 0 (16KB region).**		0h	RO
03	PREF	**Prefetchable:** 0: non-prefetchable (default for 32-bit) ***Note:*** hardwired to 0.		0b	RO
02 : 01	TYP	**Addressing Type:** 00 - 32-bit (default for 32-bit) ***Note:*** hardwired to 0.		00b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space. ***Note:*** hardwired to 0.		0b	RO

29.3.2.6 GbEPCIBAR4[3:0]—BAR4 Base Address Register

See Table 29-18 and Table 29-19. This register is the lower 32-bit of a 64-bit BAR. This register is reserved when GbE is setup to use 32-bit BARs. BARCTRL.BAR32 is used as the qualifier signal.

Table 29-23. GbEPCIBAR4[0:3]—BAR4 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 14	ADDR	**Programmable Base Address:** These bits are set by BIOS to locate the base address of the region. - RW for 64-bit BARs. - RV for 32-bit BARs. ***Note:*** BARCTRL.BAR32 controls these bits attributes.		0h	RV
13 : 04	RES	**Hardwired to 0 (16KB region).**		0h	RO
03	PREF	**Prefetchable:** 0: non-prefetchable (default for 32-bit) - not applicable as this register is not used in 32-bit mode. 1: prefetchable (default for 64-bit) ***Note:*** BARCTRL.BAR32 controls reset value. When BARCTRL.BAR32=0 then PREF=BARCTRL.PREFBAR otherwise PREF=0.		0	RO
02 : 01	TYP	**Addressing Type:** 00 - 32-bit (default for 32-bit)- not applicable as this register is not used in 32-bit mode. 10 - 64-bit (default for 64-bit) ***Note:*** BARCTRL.BAR32 controls reset value.		00b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

29.3.2.7 GbEPCIBAR5[0:3]—BAR5 Base Address Register

See Table 29-18 and Table 29-19. This register is the upper 32-bit of a 64-bit BAR. This register is reserved when GbE is setup to use 32-bit BARs. BARCTRL.BAR32 is used as the qualifier signal.

Table 29-24. GbEPCIBAR5[0:3]—BAR5 Base Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 00	ADDRRES	- Reserved for 32-bit (RV for 32-bit BARs) - Upper Address for 64-bit (RW for 64-bit BARs) ***Note:*** BARCTRL.BAR32 controls these bits attributes.		0h	RV

29.3.2.8 PSVID[0:3]—PF Subsystem Vendor ID Register

This value can be loaded automatically from EEPROM address 0x0C at power up or reset. A value of 0x8086 is the default for this field at power up if the EEPROM does not respond or is not programmed. All functions are initialized to the same value.

Table 29-25. PSVID[0:3]—PF Subsystem Vendor ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	SVID	**Subsystem Vendor ID:** This value can be loaded automatically from EEPROM address 0x0C at power up or reset. A value of 0x8086 is the default for this field at power up if the EEPROM does not respond or is not programmed.		8086h	RO

29.3.2.9 PSID[0:3]—PF Subsystem ID Register

Default to 0h. Value can be loaded fro EEPROM offset 0x0B.

Table 29-26. PSID[0:3]—PF Subsystem ID Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	SID	**Subsystem ID:** Vendor supplied device ID. Default is 0h and identifies the device as Intel.23‘. Loaded from EEPROM in reset.		0h	RO

29.3.2.10 Expansion ROM Base Address (0x30; RO) (Flash Not Implemented)

29.3.2.11 PCP[0:3]—PF Capabilities Pointer Register

Table 29-27. PCP[0:3]—PF Capabilities Pointer Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 40h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	CP	**Pointer to First Capability Structure:** Value points to the configuration space offset of the first capability structure (MSI).		40h	RO

The *Capabilities Pointer* field (Cap_Ptr) is an 8-bit field that provides an offset in the device's PCI configuration space for the location of the first item in the Capabilities Linked List (CLL). The Controller sets this bit and implements a capabilities list to indicate that it supports PCI power management, Message Signaled Interrupts (MSIs), and PCIe* extended capabilities. Its value is 0x40, which is the address of the first entry: PCI power management.

29.3.2.12 PIRQL[0:3]—PF Interrupt Line Register

Table 29-28. PIRQL[0:3]—PF Interrupt Line Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	IRQL	**Interrupt Line:** BIOS writes the interrupt routing information to this register to indicate which input of the interrupt controller this device is connected to. The device itself does not use this information.		0h	RW

29.3.2.13 PIRQP0—PF Interrupt Pin Register 0

Read only register. For PCH implementation, interrupt mapping is fixed to LAN0 = INTB#, LAN1 = INTC#, LAN2 = INTD#, and LAN3 = INTB#.

Table 29-29. PIRQP0—PF Interrupt Pin Register 0

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 02h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	IRQP	**Interrupt Pin:** Set as follows: PIRQ[0]/LAN0 = 02h (INTB#) PIRQ[1]/LAN1 = 03h (INTC#) PIRQ[2]/LAN2 = 04h (INTD#) PIRQ[3]/LAN3 = 02h (INTB#)		02h	RO

29.3.2.14 PIRQP1—PF Interrupt Pin Register 1

Read only register. For PCH implementation, interrupt mapping is fixed to LAN0 = INTB#, LAN1 = INTC#, LAN2 = INTD#, and LAN3 = INTB#.

Table 29-30. PIRQP1—PF Interrupt Pin Register 1

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 03h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	IRQP	**Interrupt Pin:** Set as follows: PIRQ[0]/LAN0 = 02h (INTB#) PIRQ[1]/LAN1 = 03h (INTC#) PIRQ[2]/LAN2 = 04h (INTD#) PIRQ[3]/LAN3 = 02h (INTB#)		03h	RO

29.3.2.15 PIRQP2—PF Interrupt Pin Register 2

Read only register. For PCH implementation, interrupt mapping is fixed to LAN0 = INTB#, LAN1 = INTC#, LAN2 = INTD#, and LAN3 = INTB#.

Table 29-31. PIRQP2—PF Interrupt Pin Register 2

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 04h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	IRQP	**Interrupt Pin:** Set as follows: PIRQ[0]/LAN0 = 02h (INTB#) PIRQ[1]/LAN1 = 03h (INTC#) PIRQ[2]/LAN2 = 04h (INTD#) PIRQ[3]/LAN3 = 02h (INTB#)		04h	RO

29.3.2.16 PIRQP3—PF Interrupt Pin Register 3

Read only register. For PCH implementation, interrupt mapping is fixed to LAN0 = INTB#, LAN1 = INTC#, LAN2 = INTD#, and LAN3 = INTB#.

Table 29-32. PIRQP3—PF Interrupt Pin Register 3

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 02h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	IRQP	**Interrupt Pin:** Set as follows: PIRQ[0]/LAN0 = 02h (INTB#) PIRQ[1]/LAN1 = 03h (INTC#) PIRQ[2]/LAN2 = 04h (INTD#) PIRQ[3]/LAN3 = 02h (INTB#)		02h	RO

29.4 PCI Capabilities

The first entry of the PCI capabilities link list is pointed by the Cap_Ptr register. The following table describes the capabilities supported by the Controller.

Address	Item	Next Pointer
0x40-47	PCI Power Management	0x50
0x50-67	Message Signaled Interrupt	0x70
0x70-8B	Extended Message Signaled Interrupt	0xA0
0xA0-DB	PCIe* Capabilities	0xE0/0x00[1]
0xE0-0xE7	Vital Product Data Capability	0x00

1. Next pointer is 0x00 if the VPD area in the EEPROM does not exist. In EEPROM-less mode, the PCIe* capability is the last capabilities section.

29.4.1 Power Management Capability Structure

All fields are reset on full power-up. All of the fields except *PME_En* and *PME_Status* are reset on exit from D3cold state. If aux power is not supplied, the *PME_En* and *PME_Status* fields also reset on exit from D3cold state.

29.4.1.1 PPMCAP[0:3]—PF Power Management Capabilities ID Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe* Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 29-33. PPMCAP[0:3]—PF Power Management Capabilities ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 40h **Offset End:** 40h	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PMID	**Capability ID:** PCI SIG assigned capability record ID (01h, Power Management capability)		01h	RO

29.4.1.2 PPMCP[0:3]—PF Power Management Next Capability Pointer Register

Table 29-34. PPMCP[0:3]—PF Power Management Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 41h **Offset End:** 41h	
Size: 8 bit	**Default:** 50h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PMCP	**Next Capability Pointer:** PCI Express Capability.		50h	RO

29.4.1.3 PPMC[0:3]—PF Power Management Capabilities Register

This field describes the Controller's functionality at the power management states as described in the following table. Each device function has its own register.

Table 29-35. PPMC[0:3]—PF Power Management Capabilities Register

Description:						
View: PCI		**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 42h **Offset End:** 43h	
Size: 16 bit		**Default:** C823hh			**Power Well:** Core	
Bit Range		**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	:11	PME	PME_Support - This 5-bit field indicates the power states in which the function may assert PME#. A value of 0b for any bit indicates that the function is not capable of asserting the PME# signal while in that power state. bit(11) X XXX1b - PME# can be asserted from D0 bit(12) X XX1Xb - PME# can be asserted from D1 bit(13) X X1XXb - PME# can be asserted from D2 bit(14) X 1XXXb - PME# can be asserted from D3hot bit(15) 1 XXXXb - PME# can be asserted from D3cold Value of this field is a function of Aux Pwr availability and *Power Management* (PM Ena) bit in *Initialization Control Word 1* (word 0x0A) EEPROM word. **Condition** **Functionality** **Value** PM Dis in EEProm No PME at all states 00000b PM Ena & NoAux Pwr PME at D0 and D3hot 01001b PM Ena & Aux Pwr PME at D0, D3hot and D3cold 11001b		11001b	RO
	10	D2	D2_Support The Controller does not support D2 state.		0b	RO
	9	D1	D1_Support The Controller does not support D1 state.		0b	RO
8	:6	AC	AUX Current - Required current defined in the Data Register. Not supported by CC.		000b	RO
	5	DSI	DSI The Controller requires its device driver to be executed following transition to the D0 uninitialized state.		1b	RO
	4	Reserved	Reserved		0b	RO
	3	PMC	PME_Clock Disabled. Hardwired to 0b.		0b	RO
2	:0	PMV	Version The Controller complies with the PCI PM specification, revision 1.2.		11b	RO

29.4.1.4 PPMCSR—PF Power Management Control and Status Register

This register is used to control and monitor power management events in the Controller. Each device function has its own PMCSR.

Table 29-36. PPMCSR[0:3]—PF Power Management Control and Status Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** 44h **Offset End:** 47h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved/NA/Not Supported for EP.		0000h	RO
15	PMS	**PME Status**: This is driven through sideband signal. EP is required to generate PME. This bit is set to 1b when the function detects a wake-up event independent of the state of the PME_En bit. Writing a 1b clears this bit.		0b	RW1C
14 :13	DSC	**Data Scale**: Set to 0 because the EP does not implement the Data register. If the Data register has not been implemented, this field must return "00b" when the PMCSR is read.		00b	RO
12 :9	DSEL	**Data Select**: Set to 0 for EP because the EP does not implement the Data register (The Data register is an optional, 8-bit read-only register that provides a mechanism for the function to report state dependent operating data such as power consumed or heat dissipation) If the Data register is not implemented, this field should be read only and return "0000b" when the PMCSR is read.		0000b	RO
8 8	PME	**PME Enable**: This is driven through sideband signal. EP is required to generate PME.		0b	RW
7 :4	Reserved	Reserved		0000b	RV
3	NSR	**No_Soft_Reset**: This bit is always set to 0b to indicate that GbE performs an internal reset after a transition from D3hot to D0 via software control of the PowerState bits. Configuration context is lost when performing the soft reset. After transitioning from the D3hot to the D0 state, full re-initialization sequence is needed to return GbE to D0 Initialized.		0b	RO
2	Reserved	Reserved		0b	RO
1 :0	PS	**Power State**. This field is used to determine the current power state of EP and to set a new power state. • 00: D0 • 01: D1 (Not supported by EP, ignore writes with this value) • 10: D2 (Not supported by EP, ignore writes with this value) • 11: D3_hot If software selects a Power state that is not supported by the EP (D2/D1), the writes must complete normally on PCIe*, but the write data is discarded and no state change occurs. The GbE also supports D3_cold.		00b	RW

29.4.2 MSI Capability Structure

This structure is required for PCIe* devices.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x50	Message Control (0x0080)		Next Pointer (0x70)	Capability ID (0x05)
0x54	Message Address			
0x58	Message Upper Address			
0x5C	Reserved		Message Data	
0x60	Mask bits			
0x64	Pending bits			

29.4.2.1 PMSICID[0:3]—Message Signalled Interrupt Capability ID Register

The Message Signalled Interrupt Capability record defines how the device generates PCI MSI messages. It is an 10B PCI SIG-defined capability record and includes the MCID, MCP, MCTL, MADR, and MDATA fields of the configuration header.

Table 29-37. PMSICID[0:3] - Message Signalled Interrupt Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 50h **Offset End:** 50h	
Size: 8 bit	**Default:** 05h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	MCID	**Capability ID:** PCI SIG assigned capability record ID (05h, MSI capability)		05h	RO

29.4.2.2 PMSINCP[0:3]—Message Signalled Interrupt Next Capability Pointer Register

Table 29-38. PMSINCP[0:3]—Message Signalled Interrupt Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 51h **Offset End:** 51h	
Size: 8 bit	**Default:** 70h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	MCP	**Next Capability Pointer:** MSI-X capability.		70h	RO

29.4.2.3 PMSICTL[0:3]—Message Signalled Interrupt Control Register

Table 29-39. PMSICTL[0:3]—Message Signalled Interrupt Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** 0180h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 09	Reserved	Reserved		0h	RO
08	MC	**Per-Vector Masking Capable:** Hard-wired to 1. A value of 1b indicates that GbE is capable of per-vector masking.		1b	RO
07	C64	**64 bit Address Capable:** Hardwired to 1 to indicate the device generate 64b message addresses.		1b	RO
06 : 04	MME	**Multiple Message Enable:** System software writes to this field to indicate the number of allocated messages (less than or equal to the number of requested messages in MMC). A value of 0 corresponds to one message.		000h	RW
03 : 01	MMC	**Multiple Message Capable:** System software reads this field to determine the number of requested messages. Hardwired to 0 to request one message.		000h	RO
00	MSIE	MSI Enable: System software sets this bit to enable MSI signaling. A device driver is prohibited from writing this bit to mask a device's service request. If 1, the device can use an MSI to request service. If 0, the device cannot use an MSI to request service.		0b	RW

29.4.2.4 PMSILADDR[0:3]—Message Signalled Interrupt Lower Address Register

Table 29-40. PMSILADDR[0:3]—Message Signalled Interrupt Lower Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	**Lower Message Address:** Written by the system to indicate the lower 30-bits of the address to use for the MSI memory write transaction. The lower two bits will always be written as 0.		0h	RW
01 :00	Reserved	**Reserved**		00b	RV

29.4.2.5 PMSIUADDR[0:3]—Message Signalled Interrupt Upper Address Register

Table 29-41. PMSIUADDR[0:3]—Message Signalled Interrupt Upper Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 58h **Offset End:** 5Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Message Address:** Written by the system to indicate the upper 30-bits of the address to use for the MSI memory write transaction.		0h	RW

29.4.2.6 PMSIDATA[0:3]—Message Signalled Interrupt Data Register

Table 29-42. PMSIDATA[0:3]—Message Signalled Interrupt Data Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 5Ch **Offset End:** 5Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 00	DATA	**Message Data:** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWORD transaction. The upper 16 bits of the transaction are written as 0.		0h	RW

29.4.2.7 PMSIMSK[0:3]—Message Signalled Interrupt Mask Register

The Mask Bits and Pending Bits registers enable software to disable or defer message sending on a per-vector basis. As the Controller supports only one message, only bit 0 of these register is implemented.

Table 29-43. PMSIMSK[0:3]—Message Signalled Interrupt Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 60h **Offset End:** 63h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 01	Reserved	**Reserved**		0h	RV
00 : 00	MSK0	MSI Vector 0 Mask If set, the Controller is prohibited from sending MSI messages.		0h	RW

29.4.2.8 PMSIPND[0:3]—Message Signalled Interrupt Mask Pending Register

Table 29-44. PMSIPND[0:3]—Message Signalled Interrupt Pending Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 64h **Offset End:** 67h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 01	Reserved	**Reserved**		0h	RV
00 : 00	PND0	If set, the Controller has a pending MSI message.		0h	RO

29.4.3 MSI-X Configuration

More than one MSI-X capability structure per function is prohibited, but a function is permitted to have both an MSI and an MSI-X capability structure.

In contrast to the MSI capability structure, which directly contains all of the control/status information for the function's vectors, the MSI-X capability structure instead points to an MSI-X table structure and a MSI-X Pending Bit Array (PBA) structure, each residing in memory space.

Each structure is mapped by a Base Address Register (BAR) belonging to the function, located beginning at 0x10 in configuration space. A BAR Indicator Register (BIR) indicates which BAR, and a Qword-aligned offset indicates where the structure begins relative to the base address associated with the BAR. The BAR is permitted to be either 32-bit or 64-bit, but must map to memory space. A function is permitted to map both structures with the same BAR, or to map each structure with a different BAR.

The MSI-X table structure, typically contains multiple entries, each consisting of several fields: message address, message upper address, message data, and vector control. Each entry is capable of specifying a unique vector.

The PBA structure, described in the same section, contains the function's pending bits, one per Table entry, organized as a packed array of bits within Qwords. The last Qword might not be fully populated.

To request service using a given MSI-X table entry, a function performs a Dword memory write transaction using:

- The contents of the Message Data field entry for data.
- The contents of the Message Upper Address field for the upper 32 bits of the address.
- The contents of the Message Address field entry for the lower 32 bits of the address.

A memory read transaction from the address targeted by the MSI-X message produces undefined results.

The MSI-X table and MSI-X PBA are permitted to co-reside within a naturally aligned 4 KB address range, though they must not overlap with each other.

MSI-X table entries and Pending bits are each numbered 0 through N-1, where N-1 is indicated by the Table Size field in the MSI-X Message Control register. For a given arbitrary MSI-X table entry K, its starting address can be calculated with the formula:

Entry starting address = Table base + K*16

For the associated Pending bit K, its address for Qword access and bit number within that Qword can be calculated with the formulas:

Qword address = PBA base + (K div 64)*8

Qword bit# = K mod 64

Software that chooses to read Pending bit K with Dword accesses can use these formulas:

Dword address = PBA base + (K div 32)*4

Dword bit# = K mod 32

The Controller also supports the table-less MSI-X mode, where a single interrupt vector is provided. The MSI-X table and MSI-X PBA are not used. Instead, the capability structure includes several additional fields (Message Address, Message Address Upper, and Message Data) for vector configuration. The Controller embeds the number of the original MSI-X vectors (for example, the vectors supported if the number of vectors was not limited to 1) in the LSB bits of the Message Data field.

Table 29-45. MSI-X Capability Structure

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x70	Message Control (0x00090)		Next Pointer (0xA0)	Capability ID (0x11)
0x74	Table Offset			
0x78	PBA offset			

29.4.3.1 PMSI-X[0:3]—PF Message Signalled Interrupt X Capability ID Register

MSI-X Capability ID Register indicates that the device is capable of generating MSI-X Interrupts.

Table 29-46. PMSI-X[0:3]—PF Message Signalled Interrupt X Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 70h **Offset End:** 70h	
Size: 8 bit	**Default:** 11h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	MSIX	**Capability ID:** PCI SIG assigned capability record ID (11h, MSI-X capability). 11h identifies the EP as a device that is MSI-X capable.		11h	RO

29.4.3.2 PMSIXNCP[0:3]—PF MSIX Next Capability Pointer Register

Table 29-47. PMSIXNCP[0:3]—PF MSIX Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 71h **Offset End:** 71h	
Size: 8 bit	**Default:** A0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	MCP	**Next Capability Pointer:** Power Management Capability		A0h	RO

29.4.3.3 PMSIXCNTL[0:3]—PF Message Signalled Interrupt X Control Register

Table 29-48. PMSIXCNTL[0:3]—PF Message Signalled Interrupt X Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 72h **Offset End:** 73h	
Size: 16 bit	**Default:** 0009h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	MSIXEN	MSI-X Enable: This bit enables the EP to generates interrupts using the MSI-X tables instead of the legacy INTx messages. When this bit is set to 1, the EP will not generate INTx messages and must use the MSI-X to signal interrupts. The device driver should not clear this bit to mask interrupts. This bit will be set by the system PCI device manager.		0h	RW
14	FM	Function Mask: This bit controls the masking of all vectors implemented in the EP. When this bit is 0, each vectors mask bit determines whether the vector is masked or not. When this bit is 1, all vectors are masked regardless of the per vector masking bit.		0h	RW
13 :11	Reserved	Always returns 0's when read.		0h	RO
10 :00	TS	MSI-X Table Size: Number of vectors (encoded as N-1) supported by the EP. The EP supports 10 vectors.		9h	RO

29.4.3.4 PMSIXTBIR[0:3]—PF MSI-X Table Offset & Table BIR Register

Table 29-49. PMSIXTBIR[0:3]—PF MSI-X Table Offset & Table BIR Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 74h **Offset End:** 77h	
Size: 32 bit	**Default:** 00000003h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	TO	Table Offset: Offset to a location in one of the EP's BAR's (indicated by TBIR) that points to the location of the base of the MSI-X Table. The MSI-X Table is mapped to offset 0KB of the GbE BAR3 or BAR4.		0h	RO
2 :0	TBIR	Table BAR Indicator Register: The BIR is mapped in the GbE BAR3 or BAR4. ***Note:*** Reset value controls by EEPROM.		11b	RO

29.4.3.5 PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register

Table 29-50. PMSIXPBABIR[0:3]—PF MSI-X Pending Bit Array & BIR Offset Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 00002003h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	PBAO	Pending Bit Array Offset: Offset to a location in one of the EP BAR's (indicated by PBABIR) that points to the location of the base of the MSI-X PBA. The MSI-X PBA maps to offset 8KB of GbE BAR3 or BAR4.		00000000000000000100000000000b	RO
2 : 00	PBABIR	Pending Bit Array BAR Indicator Register: The MSI-X PBA is mapped in GbE BAR3 or BAR4. ***Note:*** Reset value controls by EEPROM.		11b	RO

29.4.4 CSR Access Via Configuration Address Space

29.4.4.1 IOADDR[0:3]—IOADDR Register

Read/write register. Each function has its own *IOADDR* register. Functionality is the same in all functions. Register is cleared at Power-up (LAN_PWR_GOOD) or PCIe* reset.

Table 29-51. IOADDR[0:3]—IOADDR Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 98h **Offset End:** 9Bh	
View: PCI	**BAR:** GbEPCIBAR2		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 00h **Offset End:** 03h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31	EN	Configuration IO Access Enable. 0b - CSR configuration read or write disabled. 1b - CSR Configuration read or write enabled When bit is set accesses to the IODATA register actually generate transactions to the device. Otherwise, accesses to the IODATA register are don't-cares (write are discarded silently, reads return arbitrary results).		0b	RW1
30 :0	ADDR	Internal Register or Internal Memory location Address. 0x00000-0x1FFFF - Internal Registers and Memories 0x20000-0x7FFFFFFF - Undefined		0h	RW[1]

1. In the event that the *CSR_conf_en* bit in the *PCIe* Init Configuration 2* EEPROM word is cleared, accesses to the *IOADDR* register via configuration address space is ignored and has no effect on the register and the CSRs referenced by the *IOADDR* register.

29.4.4.2 IODATA[0:3]—IODATA Register

This is a read/write register. Each function has its own *IODATA* register. Functionality is the same in all functions. Register is cleared at Power-up (LAN_PWR_GOOD) or PCIe* reset.

Table 29-52. IODATA[0:3]—IODATA Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 9Ch **Offset End:** 9Fh	
View: PCI	**BAR:** GbEPCIBAR2		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 04h **Offset End:** 07h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :0	DATA	Data field for reads or writes to the Internal Register location as identified by the current value in IOADDR. All 32 bits of this register are read/write-able.		0h	RW[1]

1. In the event that the *CSR_conf_en* bit in the *PCIe* Init Configuration 2* EEPROM word is cleared, access to the *IODATA* register via configuration address space is ignored and has no effect on the register and the CSRs referenced by the *IOADDR* register.

29.4.5 PCIe* Configuration Registers

PCIe* provides two mechanisms to support native features:

- PCIe* defines a PCI capability pointer indicating support for PCIe*.
- PCIe* extends the configuration space beyond the 256 bytes available for PCI to 4096 bytes.

The Controller implements the PCIe* capability structure for endpoint devices as follows:

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0xA0	PCI Express Capability Register (0x0002)		Next Pointer (0xE0)	Capability ID (0x10)
0xA4	Device Capability			
0xA8	Device Status		Device Control	
0xAC	Link Capability			
0xB0	Link Status		Link Control	
0xB4	Reserved			
0xB8	Reserved		Reserved	
0xBC	Reserved			
0xC0	Reserved		Reserved	
0xC4	Device Capabilities 2			
0xC8	Reserved		Device Control 2	
0xCC	Reserved			
0xD0	Link Status 2		Link Control 2	
0xD4	Reserved			
0xD8	Reserved		Reserved	

29.4.6 PCI Express Capability Structure

29.4.6.1 PPCID[0:3]—PF PCI Express Capability ID Register

The PCI Express Capability List register enumerates the PCI Express Capability structure in the PCI 3.0 configuration space capability list.

Table 29-53. PPCID[0:3]—PF PCI Express Capability Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A0h **Offset End:** A0h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PCIECID	**Capability ID:** PCI SIG assigned capability record ID (10h, PCI Express capability)		10h	RO

29.4.6.2 PPCP[0:3]—PF PCI Express Next Capability Pointer Register

Table 29-54. PPCP[0:3]—PF PCI Express Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A1h **Offset End:** A1h	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PCIENP	**Next Capability Pointer:** Offset to the next capability item in the capability list. Its value of 0xE0 points to the VPD structure. If VPD is disabled, a value of 0x00 value indicates that it is the last item in the capability-linked list. VPD enable is a sideband signal from GbE to EP. Value loaded by EEPROM.		00h	RO

29.4.6.3 PPCR[0:3]—PF PCI Express Capabilities Register

Table 29-55. PPCR[0:3]—PF PCI Express Capabilities Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A2h **Offset End:** A3h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 : 14	Reserved	Reserved		00b	RO
13 :9	IMN	**Interrupt Message Number:** This field indicates which MSI vector is used for the interrupt message generated in association with the status bits in either the Slot Status field of this capability structure (applies to only RC or Switch) Not applicable to EP.		00000b	RO

Table 29-55. PPCR[0:3]—PF PCI Express Capabilities Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A2h **Offset End:** A3h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
8	SI	**Slot Implemented:** This bit when set indicates that the PCI Express Link associated with this port is connected to a slot. Hardwired to 0 for EP.		0b	RO
7 :4	DPT	**Device/Port Type:** Indicates the type of PCI Express logical device. Hardwired to 0000b (PCIe* Endpoint)		0000b	RO
3 :0	CV	**Capability Version:** Indicates PCI-SIG defined PCI Express capability structure version number. EP is PCIe* 2.0 Specification Compliant.		0010b	RO

29.4.6.4 PPDCAP[0:3]—PF PCI Express Device Capabilities Register

Table 29-56. PPDCAP[0:3]—PF PCI Express Device Capabilities Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved	Reserved		0h	RO
28	FLR	**Function level reset**: When set indicates that the device supports the FLR feature. When SR-IOV is enabled both PF & VF have to support FLR *Note: Loaded from Bit 10 (FLR capability enable) in PCIe* Control2 (offset 0x28)*		1b	RO
27 :26	CSPS	**Captured Slot Power Limit Scale:** Does not apply to EP.		00b	RO
25 :18	CSPV	**Captured Slot Power Limit Value:** Does not apply to EP.		0h	RO
17 :16	Reserved	Reserved		0h	RO
15	RBEP	**Role-Based Error Reporting:** Indicates that EP conforms to Role based error reporting ECN for PCIe* 1.0a and which was subsequently rolled in PCIe* 1.1 and future revisions.		1b	RO
14 :12	ATTN	**Attention Button/Indicator Present and Power Indicator Present**. None of these are implemented in the EP.		000b	RO
11 :9	EL1L	**Endpoint L1 Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. Hardcoded to the lowest value of 1us.		000b	R0
8 :6	EL0L	**Endpoint L0s Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. This value is hardcoded to the latency of 64-128ns.		001b	R0

Table 29-56. PPDCAP[0:3]—PF PCI Express Device Capabilities Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 10008041h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
5	ETFS	**Extended Tag Field Supported:** This field indicates the maximum supported size of the Tag field as a Requester. When Clear indicates 5-bit Tag field is supported. This limits the EP to maximum of 32 outstanding requests.		0b	R0
4 :3	PFS	**Phantom Functions Supported:** This field indicates the support for use of unclaimed function numbers to extend the number of outstanding transactions allowed. EP does not use this capability.		00b	RO
2 :0	MPS	**Max_Payload_Size Supported:** This field indicates the maximum payload size that EP can support for TLPs. This value is set to indicate 256B. The defined encodings are: • 000b = 128B max payload size • 001b = 256 bytes max payload size (Max supported) • 010b = 512 bytes max payload size • 011b = 1KB max payload size • 100b = 2KB max payload size • 101b = 4KB max payload size		001b	RO

29.4.6.5 PPDCNTL[0:3]—PF PCI Express Device Control Register

Table 29-57. PPDCNTL[0:3]—PF PCI Express Device Control Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A8h **Offset End:** A9h	
Size: 16 bit	**Default:** 2810h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	STARTFLR	**Initiate FLR** – Used to initiate FLR transition. A write of 1 initiates FLR transition. Since hardware must not respond to any cycles till FLR completion, the value read by software from this bit is 0.		0b	RW
14 :12	MRS	**Max_Read_Request_Size:** This field sets the maximum Read Request size for the EP as a Requester. The EP is capable for generating up to 1kB read requests. However requests generated by the EP will be limited by the programmed value in this field. Defined encodings for this field are: 000b 128B max read request size 001b 256B max read request size 010b 512B max read request size (Default) 011b 1024B max read request size 100b 2048B max read request size 101b 4096B max read request size		10b	RW
11	ENS	**Enable No Snoop (NS):** If this bit is set to 1, EP is permitted to set the No Snoop bit in the Requester Attributes of transactions it initiates. When clear all transactions will have the No Snoop bit clear. Setting this bit will not cause the EP to set the No Snoop attribute on every memory requests that it initiates.		1b	RW
10	APME	**Auxiliary (AUX) Power PM Enable:** This bit when set enables a device to draw AUX power independent of PME AUX power.		0b	RWS
9	PFE	**Phantom Functions Enable:** When set, this bit enables a device to use unclaimed functions as Phantom Functions. Does not apply to EP.		0b	RO
8	ETFE	**Extended Tag Field Enable:** When set, this bit enables a device to use an 8-bit Tag field as a requester. Does not to EP.		0b	RO
7 :5	MPS	**Max_Payload_Size:** This field sets maximum TLP payload for EP. As a Receiver, the EP must handle TLPs as large as the set value; as a Transmitter, the EP must not generate TLPs exceeding the set value. The EP is capable of generating up to 265B MPS. However requests generated by the EP will be limited by the programmed value in this field. It is expected that the root complexes that the EP will be paired up with will be limited to 128B and 256B MPS. 000b = 128B (Default) 001b = 256B 010b = 512B 011b = 1kB Others values not supported by EP		000b	RW

Table 29-57. PPDCNTL[0:3]—PF PCI Express Device Control Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** A8h **Offset End:** A9h	
Size: 16 bit	**Default:** 2810h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
4	ERO	**Enable Relaxed Ordering:** If this bit is set, EP is permitted to set the Relaxed Ordering bit in the Attributes field of transactions it initiates. Setting this bit does not cause the EP to set the RO on every transaction it issues		1b	RW
3	URRO	**Unsupported Request Reporting Enable**: This bit, in conjunction with other bits, controls the signaling of Unsupported Requests by sending Error Messages to the root port. When clear it disables sending of error messages.		0b	RW
2	FERE	**Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_FATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
1	NERE	**Non-Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_NONEATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
0	CERE	**Correctable Error Reporting Enable:** This bit, in conjunction with other bits, controls sending ERR_COR Messages to the root port. When clear disables sending of error messages.		0h	RW

29.4.6.6 PPDSTAT[0:3]—PF PCI Express Device Status Register

Table 29-58. PPDSTAT[0:3]—PF PCI Express Device Status Register (Sheet 1 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** AAh **Offset End:** ABh	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :6	Reserved			0h	RO
5	TP	**Transactions Pending:** This bit when set indicates that EP has issued Non-Posted Requests which have not been completed either with a completion packet or completion timeout mechanism.		0h	RO
4	APD	**AUX Power Detected:** Devices that require AUX power report this bit as set if AUX power is detected by the device. Default value is set based on an external pin strap. In addition an EEPROM bit can gate this advertisement on a per function basis.		1b	RO
3	URD	**Unsupported Request Detected:** This bit indicates that EP received an Unsupported Request. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C

Table 29-58. PPDSTAT[0:3]—PF PCI Express Device Status Register (Sheet 2 of 2)

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** AAh **Offset End:** ABh	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
2	FED	**Fatal Error Detected:** This bit indicates status of Fatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C
1	NED	**Non-Fatal Error Detected:** This bit indicates status of Nonfatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C
0	CED	**Correctable Error Detected:** This bit indicates status of correctable errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW1C

29.4.6.7 PLCAPR[0:3]—PF Link Capabilities Register

Table 29-59. PLCAPR[0:3]—PF Link Capabilities Register (Sheet 1 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** ACh **Offset End:** AFh	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PORTNUM	**Port Number**: Assigned by EP after link training phase.		0h	RWOS
23 :22	Reserved	Reserved		00b	RO
21 :21	LBN	**Link Bandwidth Notification** Capability: Does not apply to endpoints/EP		0b	RO
20 :18	NA	Does not apply to EP		000b	RO
17 :15	L1EL	**L1 Exit Latency**- Indicates the exit latency from L1 to L0 state. EP does not support L1 transition 000b - Less than 1 is 001b - 1 is - 2 is 010b - 2 is - 4 is 011b - 4 is - 8 is 100b - 8 is - 16 is 101b - 16 is - 32 is 110b - 32 is - 64 is 111b - L1 transition not supported		111b	RO

Table 29-59. PLCAPR[0:3]—PF Link Capabilities Register (Sheet 2 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** ACh **Offset End:** AFh	
Size: 32 bit	**Default:** 3B502h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
14 :12	L0EL	**L0s Exit Latency**- Indicates the exit latency from L0s to L0 state. 000b - Less than 64ns 001b - 64ns – 128ns 010b - 128ns – 256ns 011b - 256ns - 512ns 100b - 512ns - 1 is 101b - 1 is – 2 is 110b - 2 is – 4 is 111b - Reserved		011b	RO
11 :10	ASLPM	**Active State Link PM Support** - Indicates the level of active state power management supported in EP. Defined encodings are: 00b - Reserved 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported		01b	RO
9 :4	LINKW	Max Link Width- Indicates the max link width. Relevant encoding: 000000b - Reserved 000001b - x1 000010b - x2 000100b - x4 001000b - x8 001100b - x12 010000b - x16 100000b - x32 EP value depends on SKU. However the max link width is x16.		010000b	RO
3 :0	MAXSPEED	Max Link Speed - Indicates Maximum supported Link Speed. Defined encodings are: 0001b - 2.5Gbs Link speed supported (Gen 1) 0010b - 5.0Gbs Link speed supported (Gen 2) EP indicates a max Link Speed of 5.0 Gbs.		0010b	RO

29.4.6.8 PLCNTLR[0:3]—PF Link Control Register

Table 29-60. PLCNTLR[0:3]—PF Link Control Register (Sheet 1 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** B0h **Offset End:** B1h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved		0000b	RO
11	LBWINTE	**Link Autonomous Bandwidth Interrupt Enable** – When set to 1b this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been set. Not applicable to EP.		0b	RO
10	LBWMINTE	**Link Bandwidth Management Interrupt Enable** – When set to 1b this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been set. Not applicable to EP		0b	RO
9	WD	**Hardware Autonomous Width Disable** – When set to 1b this bit disables hardware from changing the link width for reasons other than attempting to correct unreliable link operation by reducing link width. EP will not support this feature.		0b	RO
8	ECLKPM	**Enable Clock Power Management** – not supported in EP		0b	RO
7	EXTSYNC	**Extended Sync**. When set to one, this bit forces the transmission of: - 4096 FTS ordered sets during the L0s state - followed by a single SKP ordered set prior to entering the L0 state, - as well as the transmission of 1024 TS1 ordered sets in the L1 state prior to entering the Recovery state. This mode gives external devices (for example, logic analyzers) that may be monitoring Link activity time to achieve bit and symbol lock before the Link enters the L0 or Recovery state and resumes communication. ***Note:*** This control applies to all functions.		0b	RW
6	CCLKCFG	**Common Clock Configuration** - when set indicates that EP and the root port at the other end of the link are operating with a common reference clock. A value of 0 indicates that they operating with an asynchronous clock. This parameter affects the L0s Exit Latencies. After changing the value in this bit in both components on a Link, software must trigger the Link to retrain by writing a 1b to the Retrain Link bit. ***Note:*** This control applies to all functions.		0b	RW
5	RETRAIN	**Retrain Link**: A write of 1b to this bit initiates Link retraining by directing the Physical Layer LTSSM to the Recovery state. Does not apply to endpoint/EP		0b	RO
4	LINKDIS	**Link Disable**: This bit disables the Link by directing the LTSSM to the Disabled state when set to 1b. Does not apply to endpoint/EP		0b	RO

Table 29-60. PLCNTLR[0:3]—PF Link Control Register (Sheet 2 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** B0h **Offset End:** B1h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	RCB	**Read Completion Boundary (RCB)**: For PCIe* Endpoints this field is set optionally by configuration software to indicate the RCB value of the Root Port upstream from the Endpoint. – 0b = 64B – 1b = 128 byte Does not directly impact the EP.		0b	RW
2	Reserved	Reserved		0b	RO
1 :0	ASPMC	**Active State Link PM Control** - this field controls the active state PM supported on the link. Link PM functionality is determined by the lowest common denominator of all functions. Defined encodings are: 00b - PM Disabled 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported ***Note:*** This control applies to all functions.		00b	RW

29.4.6.9 PLSR[0:3]—PF Link Status Register

Table 29-61. PLSR[0:3]—PF Link Status Register (Sheet 1 of 2)

Description: Link Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 0102h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved		0b	RO
14	LBWMS	**Link Bandwidth Management Status** – not supported in EP. ***Note:*** This status is reflected in all functions.		0b	RO
13	DLACT	**Data Link Layer Link Active** - not supported in EP. ***Note:*** This status is reflected in all functions.		0b	RO
12	SCLKCFG	**Slot Clock Configuration** - When set indicates that EP uses the physical reference clock that the platform provides on the connector. This bit must be cleared if EP uses an independent clock. ***Note:*** This status is reflected in all functions.		0b	RWOS
11	LTINPROG	**Link Training** - Indicates that link training is in progress. Does not apply to EP/Endpoint. ***Note:*** This status is reflected in all functions.		0b	RO

Table 29-61. PLSR[0:3]—PF Link Status Register (Sheet 2 of 2)

Description: Link Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 0102h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
10	LTE	**Link Training Error** - Indicates that a link training error has occurred. Does not apply to EP. ***Note:*** This status is reflected in all functions.		0b	RO
9 :4	NLW	**Negotiated Link Width**: Negotiated Link Width - Indicates the negotiated width of the link. Relevant encoding for EP are: 000001b - x1 000010b - x2 000100b - x4 001000b - x8 010000b - x16 ***Note:*** This status is reflected in all functions.		010000b	RO
3 :0	NLS	**Negotiated Link Speed**: The negotiated Link Speed. Defined encodings are: 0001b - 2.5Gbs 0010b - 5.0Gbs ***Note:*** This status is reflected in all functions.		0010b	RO

29.4.6.10 PDCAPR2[0:3]—PF Device Capabilities 2 Register

Table 29-62. PDCAPR2[0:3]—PF Device Capabilities 2 Register

Description: Device Capabilities 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 00000012h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved.		0b	RV
4	CTODS	Completion Timeout Disable Supported. A value of 1b indicates support for the completion timeout disable mechanism.		1b	RO
3 :0	CTORS	Completion Timeout Ranges Supported: This field indicates support for the optional completion timeout programmability mechanism. This mechanism enables system software to modify the completion timeout value. Four time value ranges are defined: • Range A = 50 us to 10 ms • Range B = 10 ms to 250 ms • Range C = 250 ms to 4 s • Range D = 4 s to 64 s Bits are set according to the following table to show the timeout value ranges that are supported. • 0000b = Completion timeout programming not supported. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A & B. • 0110b = Ranges B & C. • 0111b = Ranges A, B & C. • 1110b = Ranges B, C & D. • 1111b = Ranges A, B, C & D. • All other values are reserved. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms		0010b	RO

29.4.6.11 PDCNTR2[0:3]—PF Device Control 2 Register

Table 29-63. PDCNTR2[0:3]—PF Device Control 2 Register

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** C8h **Offset End:** C9h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :5	Reserved	Reserved.		0b	RV
4	CTODIS	Completion Timeout Disable: When set to 1b, this bit disables the completion timeout mechanism. Software is permitted to set or clear this bit at any time. When set, the completion timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued. The default value for this bit is 0b.		0b	RW
3 :0	CTOV	Completion Timeout Value: In devices that support completion timeout programmability, this field enables system software to modify the completion timeout value. Encoding: • 0000b = Default range: 50 us to 50 ms. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. Values available if Range A (50 us to 10 ms) programmability range is supported: • 0001b = 50us to 100 us. • 0010b = 1 ms to 10 ms. Values available if Range B (10 ms to 250 ms) programmability range is supported: • 0101b = 16 ms to 55 ms. • 0110b = 65 ms to 210 ms. Values available if Range C (250 ms to 4 s) programmability range is supported: • 1001b = 260 ms to 900 ms. • 1010b = 1 s to 3.5 s. Values available if the Range D (4 s to 64 s) programmability range is supported: • 1101b = 4 s to 13 s. • 1110b = 17 s to 64 s. Values not defined are reserved. Software is permitted to change the value in this field at any time. For requests already pending when the completion timeout value is changed, hardware is permitted to use either the new or the old value for the outstanding requests and is permitted to base the start time for each request either when this value was changed or when each request was issued. The default value for this field is 0000b.		0000b	RW

29.4.6.12 PLCNTLR2[0:3]—PF Link Control 2 Register

Table 29-64. PLCNTLR2[0:3]—PF Link Control 2 Register (Sheet 1 of 2)

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:1+ Index1		**Offset Start:** D0h **Offset End:** D1h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved		0h	RV
12	CDE	Compliance De-emphasis – This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 1b -3.5 dB 0b -6 dB ***Note:*** This bit field is reserved for non-0 function.		0b	RV
11	CSOS	Compliance SOS – When set to 1b, the LTSSM is required to send SOS periodically in between the (modified) compliance patterns. ***Note:*** This Bit field is reserved for non-0 function.		0b	RV
10	EMC	Enter Modified Compliance – When this bit is set to 1b, the device transmits modified compliance pattern if the LTSSM enters Polling.Compliance state. ***Note:*** This bit field is reserved for non-0 function.		0b	RV
9 :7	TMARG	Transmit Margin – This field controls the value of the non deemphasized voltage level at the Transmitter pins. Encodings: 000b - Normal operating range 001b - 800-1200 mV for full swing and 400-700 mV for half-swing. 010b - (n-1) - Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n: 200-400 mV for full-swing and 100-200 mV for half-swing. other values - reserved ***Note:*** This bit field is reserved for non-0 function.		0b	RV
6	SDEM	Selectable De-emphasis. ***Note:*** This bit is not applicable and reserved for Endpoints.		0b	RO

Table 29-64. PLCNTLR2[0:3]—PF Link Control 2 Register (Sheet 2 of 2)

Description: Link Control 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** D0h **Offset End:** D1h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
5	HWAUTOSD	Hardware Autonomous Speed Disable. When set to 1b, this bit disables hardware from changing the link speed for reasons other than attempting to correct unreliable link operation by reducing link speed. Hard wire to 0b. ***Note:*** This bit field is reserved for non-0 function.		0b	RV
4	ENCOMP	Enter Compliance. Software is permitted to force a link to enter compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a link and then initiating a hot reset on the link. The default value of this field following a fundamental reset is 0b. ***Note:*** This bit field is reserved for non-0 function.		0b	RV
3 :0	TLNKS	Target Link Speed. This field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a link into compliance mode. Defined encodings are: 0001b = 2.5 Gb/s Target Link Speed. 0010b = 5 Gb/s Target Link Speed. All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. The default value of this field is the highest link speed supported (as reported in the Supported Link Speeds field of the Link Capabilities register). ***Note:*** This bit field is reserved for non-0 function.		0	RV

29.4.6.13 PLSR2[0:3]—PF Link Status 2 Register

Table 29-65. PLSR2[0:3]—PF Link Status 2 Register

Description: Link Status 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** D2h **Offset End:** D3h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :1	Reserved	Reserved		0h	RV
0	CDEL	Current De-emphasis Level – When the Link is operating at 5 GT/s speed, this bit reflects the level of de-emphasis. it is undefined when the Link is operating at 2.5 GT/s speed Encodings: 1b -3.5 dB 0b -6 dB ***Note:*** This bit field is reserved and not defined for this function.		0b	RO

29.4.7 Vital Product Data (VPD) Registers

The Controller supports access to a VPD structure stored in the EEPROM using the following set of registers.

Note: The VPD structure is available through all port functions. As the interface is common to all functions, accessing the VPD structure of one function while an access to the EEPROM is in process on another function can yield unexpected results.

Byte Offset	**Byte 3**	**Byte 2**	**Byte 1**	**Byte 0**
0xE0	VPD address		Next Pointer (0x00)	Capability ID (0x03)
0xE4	VPD data			

29.4.7.1 VPDCID[0:3]—VPD Capability ID Register

This field equals 0x3 indicating the linked list item as being the VPD registers.

Table 29-66. VPDCID[0:3]—VPD Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** E0h **Offset End:** E0h	
Size: 8 bit	**Default:** 3h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PCIECID	**Capability ID:** VPD		3h	RO

29.4.7.2 VPDNCP[0:3]—VPD Next Capability Pointer Register

Table 29-67. VPDNCP[0:3]—VPD Next Capability Pointer Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** E1h **Offset End:** E1h	
Size: 8 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 : 00	PCIENP	**Next Capability Pointer:** Last Capability.		0h	RO

29.4.7.3 VPDADDR[0:3]—VPD Address Register

Dword-aligned byte address of the VPD area in the EEPROM to be accessed. The register is read/write with the initial value at power-up indeterminate.

Table 29-68. VPDADDR[0:3]—VPD Address Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** E2h **Offset End:** E3h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	FLAG	Flag: A flag used to indicate when the transfer of data between the VPD Data register and the storage component completes. The Flag register is written when the VPD Address register is written. 0b = Read. Set by hardware when data is valid. 1b = Write. Cleared by hardware when data is written to the EEPROM. The VPD address and data should not be modified before the action completes.		0h	RW
14 : 00	VADDR	VPD Address. Dword-aligned byte address of the VPD area in the EEPROM to be accessed. The register is read/write with the initial value at power-up indeterminate. The two LSBs are RO as zero. This is the address relative to the start of the VPD area. As the maximal size supported is 256 bytes, bits 14:8 should always be zero.		0h	RW

29.4.7.4 VPDDATA[0:3]—VPD Data

This register contains the VPD read/write data.

Table 29-69. VPDDATA[0:3]—VPD Data

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** E4h **Offset End:** E7h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
31 :0	VPDDATA	VPD Data VPD data can be read or written through this register. The LSB of this register (at offset four in this capability structure) corresponds to the byte of VPD at the address specified by the VPD Address register. The data read from or written to this register uses the normal PCI byte transfer capabilities. Four bytes are always transferred between this register and the VPD storage component. Reading or writing data outside of the VPD space in the storage component is not allowed. In a write access, the data should be set before the address and the flag is set.		0h	RW

29.5 PCIe* Extended Configuration Space

PCIe* extended configuration space is located in a flat memory-mapped address space. PCIe* extends the configuration space beyond the 256 bytes available for PCI to 4096 bytes. The Controller decodes an additional 4-bits (bits 27:24) to provide the additional configuration space as shown in Table 29-70. PCIe* reserves the remaining four bits (bits 31:28) for future expansion of the configuration space beyond 4096 bytes.

The configuration address for a PCIe* device is computed using a PCI-compatible bus, device, and function numbers as follows.

Table 29-70. PCIe* Extended Configuration Space

31 28	27 20	19 15	14 12	11 2	1 0
0000b	Bus #	Device #	Fun #	Register Address (offset)	00b

PCIe* extended configuration space is allocated using a linked list of optional or required PCIe* extended capabilities following a format resembling PCI capability structures. The first PCIe* extended capability is located at offset 0x100 in the device configuration space. The first Dword of the capability structure identifies the capability/version and points to the next capability.

The Controller supports the following PCIe* extended capabilities.

Table 29-71. PCIe* Extended Capability Structure

Capability	Offset	Next Header
Advanced Error Reporting Capability	0x100	0x140/0x1A0[1]
Serial Number[2]	0x140	0x1A0

1. Depends on EEPROM settings enabling the Serial Number in *PCIe* Init Configuration 2* EEPROM Word.
2. Not available in EEPROM-less systems.

29.5.1 Advanced Error Reporting (AER) Capability

The PCIe* AER capability is an optional extended capability to support advanced error reporting. The following table lists the PCIe* AER extended capability structure for PCIe* devices.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x100	Next Capability Ptr. (0x140/0x1A0[1])	Version (0x1)	AER Capability ID (0x0001)	
0x104	Uncorrectable Error Status			
0x108	Uncorrectable Error Mask			
0x10C	Uncorrectable Error Severity			
0x110	Correctable Error Status			
0x114	Correctable Error Mask			
0x118	Advanced Error Capabilities and Control Register			
0x11C... 0x128	Header Log			

1. Depends on EEPROM settings enabling the Serial Number structure in *PCIe* Init Configuration 2* EEPROM Word.

29.5.2 PF Advanced Error Reporting Capability Structure

29.5.2.1 PPCIEAERCAPID[0:3]—PF PCI Express AER Capability ID Register

The PCI Express Capability List register enumerates the PCI Express AER Capability structure in the PCI 3.0 configuration space capability list.

Note: Depending on the ACC_VF_ENABLE strap the AER Capability can be the last capability in the list.

Table 29-72. PPCIEAERCAPID[0:3]—PF PCI Express AER Capability ID Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 100h **Offset End:** 103h	
Size: 32 bit	**Default:** 13810001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 : 20	PCIEAERNCP	Next PCI Express Extended Capability Pointer: The Next Capability Pointer default value will be dependent on the ACC_VF_ENABLE strap. 1 - 138h (Default) 0 - 000h (AER is the last extended capability)		138h	RO
19 :16	PCIEAERCVN	Advanced Error Capability Version Number: PCI Express Advanced Error Reporting Extended Capability Version Number.		1h	RO
15 : 00	PCIEAERCID	Advanced Error Capability ID: PCI Express Extended Capability ID indicating Advanced Error Reporting Capability.		1h	RO

29.5.2.2 PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register

Table 29-73. PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		00b	RV
20	UR	**Unsupported Request Error Status:** As a receiver, Set whenever an unsupported request is detected. The Header is logged.	Y	0b	RW1CS
19	ECRCC	**ECRC Check:** As a receiver, set when ECRC check fails. The Header is logged.	Y	0b	RW1CS
18	MTLP	**Malformed TLP**: As a receiver, set whenever a malformed TLP is detected. The Header is logged.	Y	0b	RW1CS
17	RO	**Receiver Overflow:** Set if PCI Express receive buffers overflow.	Y	0b	RW1CS

Table 29-73. PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
16	EC	**Unexpected Completion:** As a receiver, set whenever a completion is received that does not match the GbE requestor ID or outstanding Tag. The Header is logged.	Y	0b	RW1CS
15	CA	**Completer Abort:** As a completer, set whenever an internal agent signals a data abort. The header is logged.	Y	0b	RW1CS
14	CT	**Completion Timeout**: As a requester, set whenever an outbound Non Posted Request does not receive a completion within 16-32ms.	Y	0b	RW1CS
13	FCPES	**Flow Control Protocol Error Status:** Set whenever a flow control protocol error is detected. Not supported.	Y	0b	RW1CS
12	PTLPR	**Poisoned TLP Received:** As a receiver, set whenever a poisoned TLP is received from PCI Express. The header is logged. Internal queue errors are not covered by this bit, they are logged by the Configuration target of the transaction.	Y	0h	RW1CS
11 :6	Reserved	Reserved		0000b	RV
5	SDES	**Surprise Down Error:** Not supported.	Y	0b	RW1CS
4	DLPE	**Data Link Protocol Error:** Set whenever a data link protocol error is detected.	Y	0b	RW1CS
3 :00	Reserved	Reserved		0h	RV

29.5.2.3 PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register

Table 29-74. PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		00b	RV
20	UR	**Unsupported Request Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
19	ECRCC	**ECRC Check Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
18	MTLP	**Malformed TLP Error Mask**: When '1' error reporting is masked.	Y	0b	RWS

Table 29-74. PPAERUCM[0:3]—PF PCI Express AER Uncorrectable Error Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17	RO	**Receiver Overflow Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
16	EC	**Unexpected Completion Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
15	CA	**Completer Abort Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
14	CT	**Completion Time Out Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
13	FCPES	**Flow Control Protocol Error Mask**: When '1' error reporting is masked. Not supported.	Y	0b	RWS
12	PTLPR	**Poisoned TLP Received Error Mask**: When '1' error reporting is masked.	Y	0h	RWS
11 :6	Reserved	Reserved		0000b	RV
5	SDES	**Surprise Down Error:** Not supported.	Y	0b	RWS
4	DLPE	**Data Link Protocol Error Mask**: When '1' error reporting is masked.	Y	0b	RWS
03 :01	Reserved	Reserved		0000b	RV
0	Reserved	Undefined: Reserved		0b	RV

29.5.2.4 PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register

The Uncorrectable Error Severity register controls whether an individual uncorrectable error is reported as a non-fatal or fatal error. An error is reported as fatal when the corresponding error bit in the severity register is set. If the bit is cleared, the corresponding error is considered non-fatal.

Table 29-75. PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00062030h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	Reserved	Reserved		00b	RV
20	UR	**Unsupported Request Error Status Severity(URESS):**	Y	0b	RWS
19	ECRCC	**ECRC Check Severity:**	Y	0b	RWS
18	MTLP	**Malformed TLP Severity:**	Y	1b	RWS

Table 29-75. PPAERUCSEV[0:3]—PF PCI Express AER Uncorrectable Error Severity Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00062030h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17	RO	**Receiver Overflow Severity:**	Y	1b	RWS
16	EC	**Unexpected Completion Severity:**	Y	0b	RWS
15	CA	**Completer Abort Severity:**	Y	0b	RWS
14	CT	**Completion Time Out Severity:**	Y	0b	RWS
13	FCPES	**Flow Control Protocol Error Severity:** Not supported.	Y	1b	RWS
12	PTLPR	**Poisoned TLP Received Severity:**	Y	0h	RWS
11 :6	Reserved	Reserved		0000b	RV
5	SDES	**Surprise Down Error Severity:** Not supported.	Y	1b	RWS
04	DLPE	**Data Link Protocol Error Severity:**	Y	1b	RWS
3 :01	Reserved	Reserved		0000b	RV
0	Reserved	Undefined: Reserved		0b	RV

29.5.2.5 PPAERCS[0:3]—PF PCI Express AER Correctable Error Register

Table 29-76. PPAERCS[0:3]—PF PCI Express AER Correctable Error Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		00b	RV
13	ANFES	**Advisory Non-Fatal Error Status**	Y	0b	RW1CS
12	RTTS	**Replay Timer Timeout Status**: Set whenever a replay timer timeout occurs.	Y	0b	RW1CS
11 :09	Reserved	**Reserved**		0b	RV
08	RNRS	**REPLAY NUM Rollover Status**: Set whenever the replay number rolls over from 11 to 00.	Y	0h	RW1CS
07	BDLLPS	**Bad DLLP Status**: Sets this bit on CRC errors on DLLP.	Y	0000b	RW1CS
06	DLPE	**Bad TLP Status**: Sets this bit on CRC errors or sequence number out of range on TLP.	Y	0b	RW1CS
05 :01	Reserved	Reserved		00000b	RV
00	RES	**Receiver Error Status**: Set whenever the physical layer detects a receiver error.	Y	0b	RW1CS

29.5.2.6 PPAERCM[0:3]—PF PCI Express AER Correctable Error Mask Register

Table 29-77. PPAERCM[0:3]—PF PCI Express AER Correctable Error Mask Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 2000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		00b	RV
13	ANFES	**Advisory Non-Fatal Error Mask**: this bit is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.	Y	1b	RWS
12	RTTS	**Replay Timer Timeout Mask:**	Y	0b	RWS
11 :09	Reserved	**Reserved**		000b	RV
08	RNRS	**REPLAY NUM Rollover Mask:**	Y	0b	RWS
07	BDLLPS	**Bad DLLP Mask:**	Y	0b	RWS
06	DLPE	**Bad TLP Mask:**	Y	0b	RWS
05 :01	Reserved	Reserved		00h	RV
00	RES	**Receiver Error Mask:**	Y	0b	RWS

29.5.2.7 PPAERCTLCAP[0:3]—PF PCI Express AER Control and Capability Register

Table 29-78. PPAERCTLCAP[0:3]—PF PCI Express AER Control and Capability Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved		0	RV
08	ECRCCE	**ECRC Check Enable**: When set enables ECRC checking.	Y	0b	RWS
07	ECRCCC	**ECRC Check Capable**: Indicates EP is capable of checking ECRC.		0b	RO
06	ECRCGE	**ECRC Generation Enable**: When set enables ECRC generation.	Y	0b	RWS
05	ECRCGC	**ECRC Generation Capable**: Indicates the EP is capable of generating ECRC.		0b	RO
04 :00	TFEP	**The First Error Pointer:** Identifies the bit position of the first error reported in the Section 29-73, "PPAERUCS[0:3]—PF PCI Express AER Uncorrectable Error Status Register" ***Note:*** This register will not update until all bits in the ERRUNC STS register are cleared.		0	ROS

29.5.2.8 PPAERHDRLOG0[0:3]—PF PCI Express AER Header Log 0 Register

Table 29-79. PPAERHDRLOG0[0:3]—PF PCI Express AER Header Log 0 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 11Ch **Offset End:** 11Fh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW0	1st DWord of the Header for the PCI Express packet in error (HDRLOGDW0): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log for example the error pointer is rearmed to log again.		0h	RO

29.5.2.9 PPAERHDRLOG1[0:3]—PF PCI Express AER Header Log 1 Register

Table 29-80. PPAERHDRLOG1[0:3]—PF PCI Express AER Header Log 1 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 120h **Offset End:** 123h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW1	2nd DWord of the Header for the PCI Express packet in error (HDRLOGDW1): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log for example the error pointer is rearmed to log again.		0h	RO

29.5.2.10 PPAERHDRLOG2[0:3]—PF PCI Express AER Header Log 2 Register

Table 29-81. PPAERHDRLOG2[0:3]—PF PCI Express AER Header Log 2 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 124h **Offset End:** 127h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW2	3rd DWord of the Header for the PCI Express packet in error (HDRLOGDW2): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log for example the error pointer is rearmed to log again.		0h	RO

29.5.2.11 PPAERHDRLOG3[0:3]—PF PCI Express AER Header Log 3 Register

Table 29-82. PPAERHDRLOG3[0:3]—PF PCI Express AER Header Log 3 Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 128h **Offset End:** 12Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRDWLOG3	4th DWord of the Header for the PCI Express packet in error (HDRDWLOG3): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log for example the error pointer is rearmed to log again.		0h	RO

29.5.3 PF Alternative Routing-ID Extended Capability Structure

This section describes the PCI Express Extended Configuration Space registers that make up the Alternative Routing ID Extended Capability Structure.

Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. See the PCI Express* Base Specification 2.0 for the full register descriptions and additional information regarding their operation.

29.5.3.1 PARIDHDR[0:3]—PF Alternative Routing ID Capability Header

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE. This is compliant with the *PCI-SIG ECN: Alternatifve Routing-ID Interpretation (ARI)*, Updated June4,2007.

Table 29-83. PARIDHDR[0:3]—PF Alternative Routing ID Capability Header

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 138h **Offset End:** 13Bh	
Size: 32 bit	**Default:** 0001000Eh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ARINCO	Next Capability Offset - This is the last extended capability.		000h	RO
19 :16	ARICV	Capability Version - This is set to 1h for the most current version of the specification.		1h	RO
15 :00	ARICV	PCI Express Extended Capability ID - The PCI SIG has assigned 000Eh to the ARI extended capability.		000EH	RO

29.5.3.2 PFARICAP0—PF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE.

Table 29-84. PFARICAP0—PF ARI Capabilities Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1	**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0200h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. The Next Function Number is set based on fuse and EEPROM bits. However, some of the values listed below will never be observed as some fuse and EEPROM bits' combinations are not valid scenarios. 0h - If Function 1 is the highest function. 2h - If Function 2 is enabled. 3h - If Function 2 is disabled, but Function 3 is enabled. 4h - If Function 2-3 are disabled, but Function 4 is enabled.		2h	RO
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: GbE does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: GbE does not support.		0b	RO

29.5.3.3 PFARICAP1—PF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE.

Table 29-85. PFARICAP1—PF ARI Capabilities Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:2		**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0300h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. The Next Function Number is set based on fuse and EEPROM bits. However, some of the values listed below will never be observed as some fuse and EEPROM bits' combinations are not valid scenarios. 0h - If Function 2 is the highest function. 3h - If Function 3 is enabled. 4h - If Function 3 is disabled, but Function 4 is enabled.		3h	RO
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: GbE does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: GbE does not support.		0b	RO

29.5.3.4 PFARICAP2—PF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE.

Table 29-86. PFARICAP2—PF ARI Capabilities Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:3		**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0400h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. The Next Function Number is set based on fuse and EEPROM bits. However, some of the values listed below will never be observed as some fuse and EEPROM bits' combinations are not valid scenarios. 0h - If Function 3 is the highest function. 4h - If Function 4 is enabled.		4h	RO
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: GbE does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: GbE does not support.		0b	RO

29.5.3.5 PFARICAP3—PF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE.

Table 29-87. PFARICAP3—PF ARI Capabilities Register

Description:					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:4		**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0500h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. The Next Function Number is set based on fuse and EEPROM bits. However, some of the values listed below will never be observed as some fuse and EEPROM bits' combinations are not valid scenarios. 0h - If Function 4 is the highest function.		5h	RO
7 :2	Reserved	Reserved.		0h	RV
1	ACS	**ACS Functional Groups Capability**: GbE does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: GbE does not support.		0b	RO

29.5.3.6 PARIDCTL[0:3]—PF Alternative Routing ID Control Register

This register contains information associated with the Alternative Routing ID capability that is implemented in the GbE.

Table 29-88. PARIDCTL[0:3]—PF Alternative Routing ID Control Register

Description:					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:1+ Index1	**Offset Start:** 13Eh **Offset End:** 13Fh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :7	Reserved	Reserved		0h	RV
6 :4	FG	**Function Group:** Hardwired to Zero as GbE does not support Function Groups.		0b	RO
3 :2	Reserved	Reserved		0b	RV
1	ACS	**ACS Functional Groups Enable**: Hardwired to Zero as GbE does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Enable**: Hardwired to Zero as GbE does not support.		0b	RO

29.5.4 TLP Processing Hint Requester (TPH) Capability

The PCIe* TPH Requester capability is an optional extended capability to support TLP Processing Hints. This is not supported by the GbE Controller MAC.

29.5.4.1 TPH CAP ID (0x1A0; RO)

Not supported.

29.5.4.2 TPH Requester Capabilities (0x1A4; RO)

Not supported

29.5.4.3 TPH Requester Control (0x1A8; R/W)

Not supported.

29.5.4.4 TPH Steering table (0x1AC - 0x1B8; R/W)

Not supported.

29.5.4.5 Latency Tolerance Requirement Reporting (LTR) Capability

Not supported.

29.5.4.6 LTR CAP ID (0x1C0; RO)

Not supported

29.5.4.7 LTR Capabilities (0x1C4; RW)

Not supported

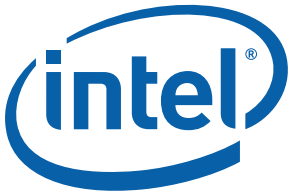
29.5.5 I/O Space

The I/O space is 32 bytes and consists of only two 32-bit registers (IOADDR and IODATA).

Table 29-89. IOBAR Register Map

Name	IOBAR Offset
IOADDR	00h
IODATA	04h
Reserved	08h-1Ch

§ §



Test Features - Volume 3 of 4

April 2014

Order Number: 327879-005US

§ §

30.0 PCH Global Test Features

30.1 JTAG

JTAG refers to the JTAG/IEEE 1149.1 Test Access Port, and is one of the primary debug and test mechanisms used on the PCH. The following subsections describe the purpose and use of the JTAG port on the PCH.

30.1.1 JTAG Functions Overview

The PCH supports the IEEE 1149.1-2001 (Standard Test Access Port and Boundary-Scan Architecture) specification. The TAP provides instructions and logic to support IEEE 1149.6-2003 (Standard for Boundary-Scan Testing of Advanced Digital Networks). The Hi-speed IOs (PCIe*, DMI,SATA*,SGMII) all support 1149.6 with the rest supporting 1149.1.

30.1.2 PCH TAP Interfaces

The PCH has two four-pin JTAG interfaces, one for the Legacy I/O (DMI) pins and the other for the pins associated with the PCI Express Endpoint. Each JTAG interface can be used independently or cascaded together as indicated in the following figure.

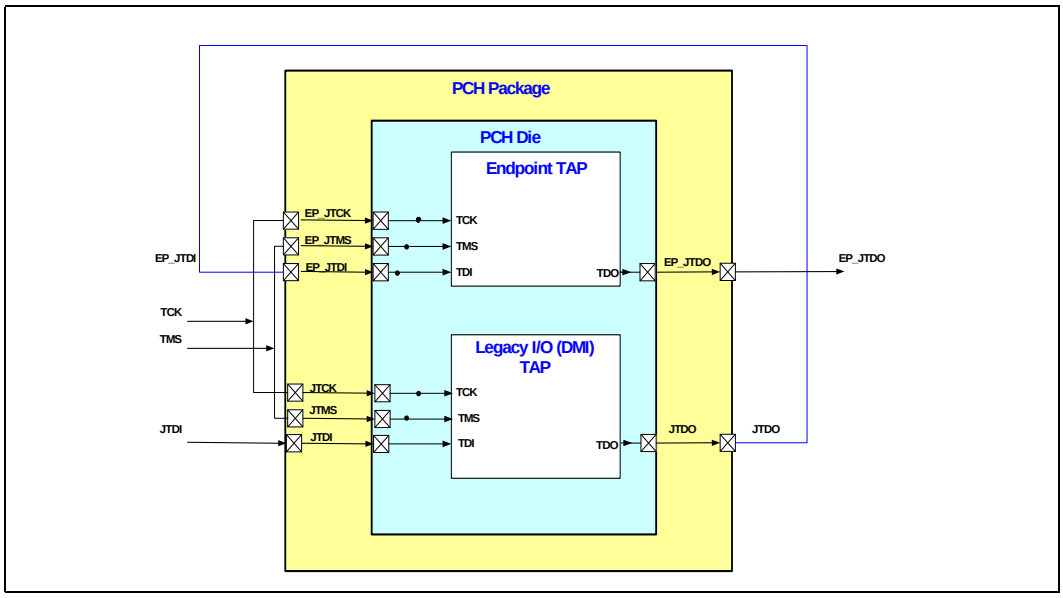
Table 30-1. PCH Die-level Legacy I/O (DMI)TAP Pin Interface

Pin Name	Type	Function	Operation and Properties
JTCK	Input	JTAG Test Clock	• Clock input for the TAP controller, instruction register and test data register.
JTMS	Input	Test Mode Select	• Changes the state of the TAP controller. • Pulled high when not being driven. • Is sampled on the rising edge of TCK.
JTDI	Input	Test Data In	• Serial data input to the instruction register and test data registers. • Pulled high when not being driven. • Is sampled on the rising edge of TCK.
JTDO	Output	Test Data Out	• Serial data output. • Inactive (high-Z) during non-shift operations. • Data is clocked out on the falling edge of TCK.

Table 30-2. PCH Die-Level Endpoint TAP Pin Interface

Pin Name	Type	Function	Operation and Properties
EP_JTCK	Input	JTAG Test Clock	• Clock input for the TAP controller, instruction register and test data register.
EP_JTMS	Input	Test Mode Select	• Changes the state of the TAP controller. • Pulled high using Internal Pull-Up when not being driven. • Is sampled on the rising edge of TCK.
EP_JTDI	Input	Test Data In	• Serial data input to the instruction register and test data registers. • Pulled high using Internal Pull-Up when not being driven. • Is sampled on the rising edge of TCK.
EP_JTRSTB	Input	Jtag Test Reset	• Jtag Reset (TRSTB) • Pulled high using Internal Pull-Up when not being driven •
EP_JTDO	Output	Test Data Out	• Serial data output. • Inactive (high-Z) during non-shift operations. Open-Drain. • Data is clocked out on the falling edge of TCK.

Figure 30-1. TAP Connectivity

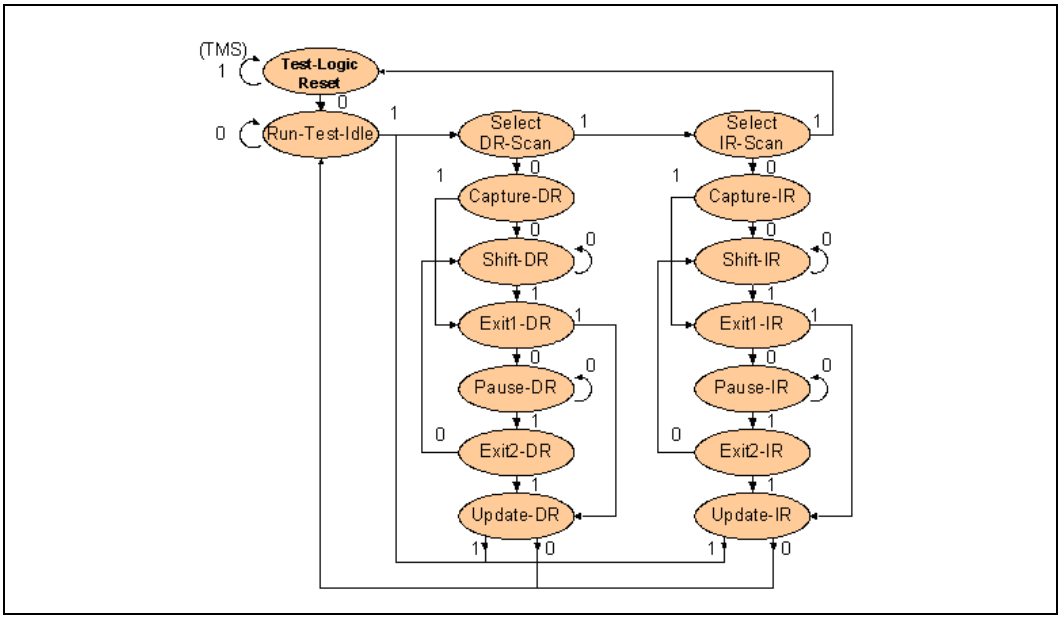


30.1.3 TAP Controller Operation and State Diagram

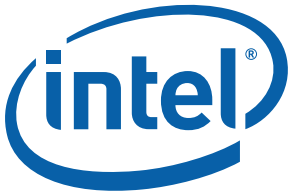
The TAP controller contains a state machine that is the heart of the JTAG TAP (Figure 30-2). The TAP controller is asynchronously reset via the hardware TAP reset. Once reset is de-asserted, the TAP controller samples the TMS pin at the rising edge of TCLK pin, and it sequences through the states under the control of the TMS pin.

Holding TMS high for five or more TCLK cycles will take the TAP to the Test-Logic-Reset state regardless of what state it is in. It is recommended that TMS be held high at the de-assertion of reset to ensure deterministic operation of the TAP. The TDO pin is output-enabled only during Shift-DR or Shift-IR states.

Figure 30-2. TAP Controller State Diagram



§ §



Technical Specifications - Volume 4 of 4

April 2014

Order Number: 327879-005US

§ §

31.0 System Clocks

31.1 External Clock Requirements

Table 31-1 shows all the input clocks to the PCH. There are three modes of operation:

- Normal - all clocks are required (all clocks listed in Table 31-1).
- EndPoint mode - only EndPoint clocks are required.
- Non-EndPoint Mode - only Non-EndPoint clocks are required.

Table 31-1. Platform External Clock Interface

Signal Name	Clock Domain	Frequency	Destination
CRU_CLK100P/N	Clock Resource Unit	100 MHz	EndPoint
GBE_CLK100P/N	GbE Ethernet	100 MHz	EndPoint
PCIE_EP_CLK100P/N	PCI Express* EndPoint	100 MHz	EndPoint
DMI_CLK100P/N	Direct Media Interface	100 MHz	Non-EndPoint
PCICLK	Low Pin Count	33 MHz	Non-EndPoint
REF_CLK14	Reference Clock	14.31818 MHz	Non-EndPoint
RTCX1 RTCX2	RTC	32.768 KHz	Non-EndPoint
SATA_CLK100P/N	Serial ATA	100 MHz	Non-EndPoint
UART_CLK	UART	48/14.7456 MHz	Non-EndPoint
USB_CLK96P/N	USB*	96 MHz	Non-EndPoint

Note: Depending on the PCH mode used for your design, all clocks may not be required. For example, in Normal mode, all interfaces are active (all EndPoint and Non-EndPoint clocks are required). The other two modes - EndPoint Mode or Non-EndPoint Mode - only require partial clock inputs. See Table 31-1 to determine which clocks are required for your mode of operation.

§ §

32.0 Signal Descriptions

This chapter provides a detailed description of each signal. The signals are arranged in functional groups according to their associated interface.

The "#" symbol at the end of the signal name indicates that the active, or asserted state occurs when the signal is at a low voltage level. When "#" is not present, the signal is asserted when at the high voltage level.

In this document:

- *Chipset*, *Platform Controller Hub*, and *PCH* are used as generic references to the Intel® Communications Chipset 89xx Series.
- Intel® Communications Chipset 89xx Series.
  - DH89xxCC includes Chipset 89xx Series SKUs where 8900 ≤ *SKU* ≤ 8920
  - DH89xxCL includes Chipset 89xx Series SKUs where 8925 ≤ *SKU* ≤ 8955

32.1 Name Convention

Table 32-1 provides the legend for interpreting the Type field that appears throughout the tables in this section.

Table 32-1. Signal Type Definitions

Type	Description
#	Active low signal
I	Input pin
O	Output pin
I/OD	Bi-directional Input/Open Drain output pin
I/O	Bi-directional Input /Output pin.
OD	Open Drain output pin
T/S	Tri-State pin
NC	No Connection to pin
RSVD	Reserved Pin. This signal must be connected as described in signal description.
CMOS	CMOS buffers

The "Type" for each signal is indicative of the functional operating mode of the signal. Unless otherwise noted, a signal is considered to be in the functional operating mode after RTCRST# de-asserts for signals in the RTC well, after RSMRST# de-asserts for signals in the suspend well, after PWROK asserts for signals in the core well, and after PCIE_EP_RST# de-asserts for signals in the EndPoint wells.

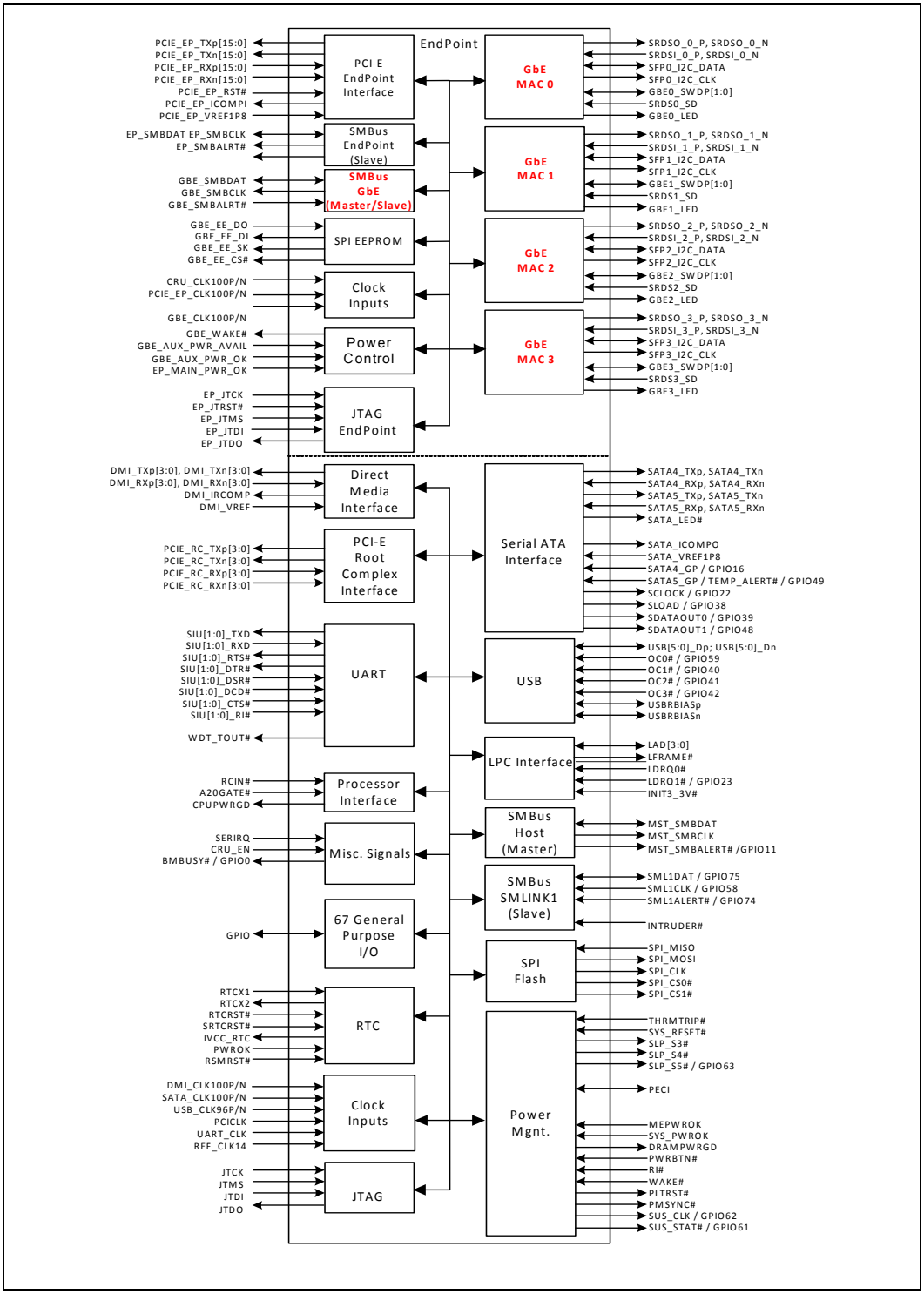
32.2 Intel® Communications Chipset 89xx Series SKU Naming

In this document:

- *Chipset*, *Platform Controller Hub*, and *PCH* are used as generic references to the Intel® Communications Chipset 89xx Series.
- Intel® Communications Chipset 89xx Series.
  - DH89xxCC includes Chipset 89xx Series SKUs where 8900 ≤ *SKU* ≤ 8920
  - DH89xxCL includes Chipset 89xx Series SKUs where 8925 ≤ *SKU* ≤ 8955

Note: In most cases, the pinlist signals apply to both DH89xxCC and DH89xxCL. Where there are differences, the pinlist differentiates the signal naming and functionality.

Figure 32-1. Interface Signals Block Diagram[1]



1. The GbE MACs (MAC 0 - MAC 3) are not available in DH89xxCL: *Intel® Communications Chipset 8900 Series, 8925 ≤ SKU ≤ 8955*. However, the EEPROM is required for EndPoint MPHY configuration.
2. The Functions in **RED** are not available in the DH89xxCL devices.

32.3 JTAG Boundary Scan Chain (BSC) 1149.1 and 1149.6 Chain

There is a total of two boundary scan chains: one is dedicated to the EndPoint and the other to the Non-EndPoint section of the chip. The PCH implements a consistent, IEEE 1149.1 compliant JTAG Boundary Scan Chain (BSC) for most interfaces enabling a low-cost manufacturing test for boards. The BSC cell is defined in the Boundary Scan Description Language (BSDL) documentation and the boundary scan is initiated by the JTAG entry.

The PCI Express* (EndPoint device) is implemented in JTAG boundary scan IEEE 1149.6. Both protocols 1149.1 and 1149.6 are implemented in a single chain, however the new features are used to validate connectivity of high speed IO pins. By toggling the pins individually or in combination, the connectivity of the pins are validated at the output pin.

32.4 Direct Media Interface

Table 32-2. Direct Media Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DMI_RXn0 DMI_RXp0	I	LV Diff	U56 T55	2		1149.1	Direct Media Interface Differential Receive Pair 0.
DMI_RXn1 DMI_RXp1	I	LV Diff	W55 W53	2		1149.1	Direct Media Interface Differential Receive Pair 1.
DMI_RXn2 DMI_RXp2	I	LV Diff	AB55 AA56	2		1149.1	Direct Media Interface Differential Receive Pair 2.
DMI_RXn3 DMI_RXp3	I	LV Diff	AC54 AC56	2		1149.1	Direct Media Interface Differential Receive Pair 3.
DMI_TXn0 DMI_TXp0	O	LV Diff	U47 U49	2		1149.1	Direct Media Interface Differential Transmit Pair 0.
DMI_TXn1 DMI_TXp1	O	LV Diff	Y47 Y49	2		1149.1	Direct Media Interface Differential Transmit Pair 1.
DMI_TXn2 DMI_TXp2	O	LV Diff	AB50 AB49	2		1149.1	Direct Media Interface Differential Transmit Pair 2.
DMI_TXn3 DMI_TXp3	O	LV Diff	AF52 AE50	2		1149.1	Direct Media Interface Differential Transmit Pair 3.
DMI_IRCOMP	I	Analog	AE47	1	External Pull-up 10.5K 1%		Direct Media Interface Impedance/Current Compensation Output. Determines DMI output impedance and bias current. Connect pull-up to VCCA1P8_DMI, 1.8V voltage supply.
DMI_VREF	I	IO Supply	AG46	1			Direct Media Interface Reference Voltage. Connect to VCCA1P8_DMI, 1.8V voltage supply.
TOTAL				18			

32.5 PCI Express* EndPoint

Table 32-3. PCI Express* Endpoint Interface Signals (Sheet 1 of 3)[1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
PCIE_EP_RXn0 PCIE_EP_RXp0	I	LV Diff	N54 N56	2		1149.1 & 1149.6	PCI Express* EndPoint Receive Data Pair Lane 0. A serial differential input pair running at a bit rate of 2.5 GT/s or 5 GT/s. The signals for this 1x16 interface can be trained to 1x4, 1x8 or 1x16 ports. (This is skew dependent; check the skew you are designing for.) These port configurations map to signals as follows (y is either "n" or "p"): TRAINING: 1X16 Interface Configuration: • PCIE_EP_RXy[15:0] 1X8 Interface Configuration: • PCIE_EP_RXy[7:0] 1X4 Interface Configuration: • PCIE_EP_RXy[3:0]
PCIE_EP_RXn1 PCIE_EP_RXp1	I	LV Diff	M55 L53	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 1
PCIE_EP_RXn2 PCIE_EP_RXp2	I	LV Diff	G54 G56	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 2
PCIE_EP_RXn3 PCIE_EP_RXp3	I	LV Diff	B51 D51	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 3
PCIE_EP_RXn4 PCIE_EP_RXp4	I	LV Diff	B47 C46	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 4
PCIE_EP_RXn5 PCIE_EP_RXp5	I	LV Diff	B45 D45	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 5
PCIE_EP_RXn6 PCIE_EP_RXp6	I	LV Diff	D43 B43	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 6
PCIE_EP_RXn7 PCIE_EP_RXp7	I	LV Diff	C42 B41	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 7
PCIE_EP_RXn8 PCIE_EP_RXp8	I	LV Diff	E41 E39	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 8
PCIE_EP_RXn9 PCIE_EP_RXp9	I	LV Diff	C39 E37	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 9
PCIE_EP_RXn10 PCIE_EP_RXp10	I	LV Diff	B37 C36	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 10
PCIE_EP_RXn11 PCIE_EP_RXp11	I	LV Diff	B35 D35	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 11
PCIE_EP_RXn12 PCIE_EP_RXp12	I	LV Diff	D33 B33	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 12

Table 32-3. PCI Express* Endpoint Interface Signals (Sheet 2 of 3)[1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
PCIE_EP_RXn13 PCIE_EP_RXp13	I	LV Diff	E31 B31	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 13
PCIE_EP_RXn14 PCIE_EP_RXp14	I	LV Diff	E29 C29	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 14
PCIE_EP_RXn15 PCIE_EP_RXp15	I	LV Diff	E27 B27	2		1149.1 & 1149.6	PCI Express EndPoint Receive Data Pair Lane 15
PCIE_EP_TXn0 PCIE_EP_TXp0	O	LV Diff	R49 R50	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 0. A serial differential output pair running at a bit rate of 2.5 GT/s or 5 GT/s. The signals for this 1x16 interface can be trained to 1x4, 1x8 or 1x16 ports. (This is skew dependent; check the skew you are designing for.) These port configurations map to signals as follows (y is either "n" or "p"): TRAINING: 1X16 Interface Configuration: • PCIE_EP_TXy[15:0] 1X8 Interface Configuration: • PCIE_EP_TXy[7:0] 1X4 Interface Configuration: • PCIE_EP_TXy[3:0]
PCIE_EP_TXn1 PCIE_EP_TXp1	O	LV Diff	R46 M46	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 1
PCIE_EP_TXn2 PCIE_EP_TXp2	O	LV Diff	M49 M48	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 2
PCIE_EP_TXn3 PCIE_EP_TXp3	O	LV Diff	K50 K49	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 3
PCIE_EP_TXn4 PCIE_EP_TXp4	O	LV Diff	H50 H48	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 4
PCIE_EP_TXn5 PCIE_EP_TXp5	O	LV Diff	J46 H46	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 5
PCIE_EP_TXn6 PCIE_EP_TXp6	O	LV Diff	L43 J43	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 6
PCIE_EP_TXn7 PCIE_EP_TXp7	O	LV Diff	J41 H41	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 7
PCIE_EP_TXn8 PCIE_EP_TXp8	O	LV Diff	P38 M38	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 8
PCIE_EP_TXn9 PCIE_EP_TXp9	O	LV Diff	H38 J38	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 9
PCIE_EP_TXn10 PCIE_EP_TXp10	O	LV Diff	H36 J36	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 10

Table 32-3. PCI Express* Endpoint Interface Signals (Sheet 3 of 3)[1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
PCIE_EP_TXn11 PCIE_EP_TXp11	O	LV Diff	M36 P36	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 11
PCIE_EP_TXn12 PCIE_EP_TXp12	O	LV Diff	L33 J33	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 12
PCIE_EP_TXn13 PCIE_EP_TXp13	O	LV Diff	P33 R33	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 13
PCIE_EP_TXn14 PCIE_EP_TXp14	O	LV Diff	H31 J31	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 14
PCIE_EP_TXn15 PCIE_EP_TXp15	O	LV Diff	M31 P31	2		1149.1 & 1149.6	PCI Express EndPoint Transmit Data Pair Lane 15
PCIE_EP_ICOMPI	I	Analog	U38	1	External Pull-up 10.5KΩ 1%		PCI Express EndPoint Impedance/Current Compensation Output. Connect pull-up to VCCAEP1P8_PE, 1.8V voltage supply.
PCIE_EP_RST#	I	LVTTL	F10	1			PCI Express Fundamental Reset. This is the warm reset for the EndPoint. On the platform it will be tied to either the PCH output PLTRST# or the PCH input SYS_RESET#.
PCIE_EP_VREF1P8	I	IO Supply	Y44	1			PCI Express EndPoint Reference Voltage. Connect to VCCAEP1P8_PE, 1.8V voltage supply.
TOTAL				67			

1. All transmit and receive signals require AC coupling capacitors. See the PDG for capacitor values and routing guidelines.

32.6 Gigabit Ethernet Interface

Note: The GbE MACs are not available in the DH89xxCL SKUs, however the EEPROM is required for Endpoint MPHY configuration.

- All Reserved (RSVDx) signals are No Connects (NC), unless specified otherwise.
- See the *Supported Ethernet PHY Devices for the Intel® Communications Chipset 89xx Series Application Note* for more information on the PHY devices supported.

Table 32-4. SerDes/SGMII Interface Signals (Sheet 1 of 3)[1]

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
SRDSI_0_N SRDSI_0_P	RSVD103 RSVD99	I	LV Diff	AJ5 AG5	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data input pair Port 0: • Differential SerDes Receive interface. A Serial differential input pair with a data rate of 1.25 Gb/s, and an embedded clock operating at 625 MHz. The embedded clock present in this input is recovered along with the data. • **DH89xxCL:** • Reserved • No Connect
SRDSI_1_N SRDSI_1_P	RSVD104 RSVD100	I	LV Diff	AL5 AJ3	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data input pair Port 1. • **DH89xxCL:** • Reserved • No Connect
SRDSI_2_N SRDSI_2_P	RSVD105 RSVD101	I	LV Diff	AM3 AL2	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data input pair Port 2. • **DH89xxCL:** • Reserved • No Connect
SRDSI_3_N SRDSI_3_P	RSVD106 RSVD102	I	LV Diff	AN4 AN2	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data input pair Port 3. • **DH89xxCL:** • Reserved • No Connect

Table 32-4. SerDes/SGMII Interface Signals (Sheet 2 of 3)[1]

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
SRDSO_0_N SRDSO_0_P	RSVD95 RSVD91	O	LV Diff	AA5 W3	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data output pair Port 0: • Differential SerDes Transmit interface. A serial differential output pair with a data rate of 1.25 Gb/s, and a clock operating at 625 MHz. This output carries both data and an embedded clock that is recovered along with data at the receiving end. • **DH89xxCL:** • Reserved • No Connect
SRDSO_1_N SRDSO_1_P	RSVD96 RSVD92	O	LV Diff	AB3 AA2	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data output pair Port 1. • **DH89xxCL:** • Reserved • No Connect
SRDSO_2_N SRDSO_2_P	RSVD97 RSVD93	O	LV Diff	AC4 AC2	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data output pair Port 2. • **DH89xxCL:** • Reserved • No Connect
SRDSO_3_N SRDSO_3_P	RSVD98 RSVD94	O	LV Diff	AB12 AB11	2		1149.1 & 1149.6	**DH89xxCC:** • SerDes/SGMII Serial Data output pair Port 3. • **DH89xxCL:** • Reserved • No Connect
SRDS0_SD	RSVD132	I	LVTTL	M3	1		1149.1	**DH89xxCC:** • Signal Detect Port 0. Indicates that signal (light) is detected from the fiber. • High = signal detect. • Low = otherwise. • **DH89xxCL:** • Reserved • No Connect

Table 32-4. SerDes/SGMII Interface Signals (Sheet 3 of 3)[1]

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
SRDS1_SD	RSVD133	I	LVTTL	C6	1		1149.1	**DH89xxCC:** • Signal Detect Port 1. Indicates that signal (light) is detected from the fiber. • High = signal detect. • Low = otherwise. • **DH89xxCL:** • Reserved • No Connect
SRDS2_SD	RSVD134	I	LVTTL	F3	1		1149.1	**DH89xxCC:** • Signal Detect Port 2. Indicates that signal (light) is detected from the fiber. • High = signal detect • Low = otherwise. • **DH89xxCL:** • Reserved • No Connect
SRDS3_SD	RSVD135	I	LVTTL	M9	1		1149.1	**DH89xxCC:** • Signal Detect Port 3. Indicates that signal (light) is detected from the fiber. • High = signal detect • Low = otherwise. • **DH89xxCL:** • Reserved • No Connect
GBE_IRCOMP	RSVD109	I	Analog	AG8	1	External Pull-up 10.5KΩ 1%		**DH89xxCC:** • GbE Impedance/Current Compensation Output. • Connect pull-up to VCCAEP1P8AUX • **DH89xxCL:** • Reserved • No Connect
GBE_VREF1P8	RSVD110	I	IO Supply	AE11	1			**DH89xxCC:** • GbE Reference Voltage. • Connect to VCCAEP1P8AUX. • **DH89xxCL:** • Reserved • No Connect
TOTAL					22			

1. All Serial Data Input signals require AC coupling capacitors. Consult the PDG for capacitor values and routing guidelines. Signal Detect Port signal must be pull down with a 10K resistor if not being used.

Table 32-5. SFP Interface Signals (Sheet 1 of 3)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		I/O Type	Tech nology	Ball Map	Ball Count	Internal/ External Resistor Pull-Up/ Down	BScan Support	Description
DH89xxCC	DH89xxCL							
SFP0_I2C_CLK	RSVD116	O,OD	LVTTL	T3	1	External pull-up required[1]	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 0 I2C Clock. • Two Wire Serial Interface (TWSI) Clock Signal. Connects to Mod-Def1 input of SFP (OD). Can also be used as MDC pin (Out). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect
SFP0_I2C_DATA	RSVD117	I/OD	LVTTL	U2	1	External pull-up required	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 0 I2C Data. • Two Wire Serial Interface (TWSI) Data Signal. Connects to Mod-Def2 input of SFP (I/OD). Can also be used as MDIO pin (I/OD). Only one of this modes can be selected at boot with EEPROM. • External pull-up required. • **DH89xxCL:** • Reserved • No Connect
SFP1_I2C_CLK	RSVD118	O,OD	LVTTL	N2	1	External pull-up required.	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 1 I2C Clock. • Two Wire Serial Interface (TWSI) Clock Signal. Connects to Mod-Def1 input of SFP (OD). Can also be used as MDC pin (Out). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect

Table 32-5. SFP Interface Signals (Sheet 2 of 3)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
SFP1_I2C_DATA	RSVD119	I/OD	LVTTL	Y8	1	External pull-up required.	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 1 I2C Data. • Two Wire Serial Interface (TWSI) Data Signal. Connects to Mod-Def2 input of SFP (I/OD). Can also be used as MDIO pin (I/OD). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect
SFP2_I2C_CLK	RSVD120	O,OD	LVTTL	R8	1	External pull-up required	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 2 I2C Clock. • Two Wire Serial Interface (TWSI) Clock Signal. Connects to Mod-Def1 input of SFP (OD). Can also be used as MDC pin (Out). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect
SFP2_I2C_DATA	RSVD121	I/OD	LVTTL	U8	1	External pull-up required.	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 2 I2C Data. • Two Wire Serial Interface (TWSI) Data Signal. Connects to Mod-Def2 input of SFP (I/OD). Can also be used as MDIO pin (I/OD). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect

Table 32-5. SFP Interface Signals (Sheet 3 of 3)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
SFP3_I2C_CLK	RSVD122	O,OD	LVTTL	L2	1	External pull-up required.	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 3 I2C Clock. • Two Wire Serial Interface (TWSI) Clock Signal. Connects to Mod-Def1 input of SFP (OD). Can also be used as MDC pin (Out). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect
SFP3_I2C_DATA	RSVD123	I/OD	LVTTL	N4	1	External pull-up required	1149.1	**DH89xxCC:** Small Form-Factor Pluggable Port 3 I2C Data. • Two Wire Serial Interface (TWSI) Data Signal. Connects to Mod-Def2 input of SFP (I/OD). Can also be used as MDIO pin (I/OD). Only one of this modes can be selected at boot with EEPROM. • External pull-up required • **DH89xxCL:** • Reserved • No Connect
TOTAL					8			

1. See the platform design collateral for information on resistor values.

Table 32-6. GbE EEPROM SPI Interface Signals

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
GBE_EE_CS#	EP_EE_CS#[1]	O	LVTTL	R12	1		1149.1	**DH89xxCC**: • GbE EEPROM Chip Select. • Chip Select output to GbE EEPROM. • **DH89xxCL:** • EEPROM Chip Select. • Chip Select output to EEPROM. • Pre-programmed EEPROM is required.[1]
GBE_EE_DI	EP_EE_DI[1]	O	LVTTL	D5	1		1149.1	**DH89xxCC**: • GbE EEPROM Data Input. • Data is output to EEPROM. • **DH89xxCL:** • EEPROM Chip Data Input. • Data is output to EEPROM. • Pre-programmed EEPROM is required.[1]
GBE_EE_DO	EP_EE_DO[1]	I	LVTTL	F8	1		1149.1	**DH89xxCC**: • GbE EEPROM Data Output. • Data is input from GbE EEPROM. • **DH89xxCL:** • EEPROM Chip Data Output. • Data is input from EEPROM. • Pre-programmed EEPROM is required.[1]
GBE_EE_SK	EP_EE_SK[1]	O	LVTTL	J10	1		1149.1	**DH89xxCC**: • GbE EEPROM Serial Clock. • Serial clock to GbE EEPROM operates at ~2 MHz. • **DH89xxCL** • EEPROM Serial Clock. • Serial clock to EEPROM operates at ~2 MHz. • Pre-programmed EEPROM required[1]
TOTAL					4			

Notes:
1. A pre-programmed EEPROM is required. There is NO WAY to update the EEPROM from an application running on IA

Table 32-7. GbE SMBus Interface Signals (Master/Slave)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
GBE_SMBCLK	RSVD114	I/OD	LVTTL	Y12	1	External pull-up required[1]	1149.1	**DH89xxCC:** GbE SMBus Clock. • One clock pulse is generated for each data bit transferred. • External pull-up resistor to VCCPEP3P3AUX is required. • Pull-up resistor value should be calculated based on the bus load, see the platform design collateral. • **DH89xxCL:** • Reserved • External pull-up resistor to VCCPEP3P3AUX is required.
GBE_SMBDAT	RSVD113	I/OD	LVTTL	U11	1	External pull-up required	1149.1	**DH89xxCC:** GbE SMBus Data. • Stable during the high period of the clock (unless it is a start or stop condition). • External pull-up resistor to VCCPEP3P3AUX is required. • Resistor value should be calculated based on the bus load, see the platform design collateral. • **DH89xxCL:** • Reserved • External pull-up resistor to VCCPEP3P3AUX is required.
GBE_SMBALRT#	RSVD115	I/OD	LVTTL	U12	1	External pull-up required	1149.1	**DH89xxCC:** GbE SMBus Alert. • Acts as an interrupt of a slave device on SMBus. • External pull-up resistor to VCCPEP3P3AUX is required. • Resistor value should be calculated based on the bus load, see the platform design collateral. • **DH89xxCL:** • Reserved • External pull-up resistor to VCCPEP3P3AUX is required.
TOTAL					3			

1. See the platform design collateral for resistor values.

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 1 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
EP_MAIN_PWR_OK		I	LVTTL	L5	1			**DH89xxCL & DH89xxCC:** EndPoint Main Power OK. • Indicates that Main Power Wells are up. • This signal can be connected to PWROK.
GBE_AUX_PWR_AVAIL	RSVD137	I	LVTTL	B7	1	External pull-up required	1149.1	**DH89xxCC:** Auxiliary Power Available. • This pin is a strapping option pin, latched at the rising edge of PCIE_EP_RST# or In-Band PCIe* Reset. If this pin is driven high during initialization time, it indicates that Auxiliary Power is available and the device should support D3COLD power state if enabled to do so. This pin value should be established before GBE_AUX_POWER_OK goes high. • **DH89xxCL:** • Reserved • No Connect
GBE_AUX_PWR_OK	RSVD138	I	LVTTL	K8	1			**DH89xxCC:** Auxiliary Power OK. • Power good reset for AUX well. • Indicates power to AUX well is stable. • **DH89xxCL:** • Reserved • No Connect
GBE_WAKE#	RSVD136	OD	LVTTL	H12	1	External pull-up required	1149.1	**DH89xxCC:** Wake on LAN. • Wake Output signal for wake on LAN. • **DH89xxCL:** • Reserved • No Connect

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 2 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE0_LED	EP_SMB_ADR2	O	LVTTL	N6	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK or EP_MAIN_PWR_OK See ***Note*** in the DH89xxCC Description column	1149.1	**DH89xxCC:** • Port 0 LED. Programmable LED, mode encoding set by GbE EEPROM. • On power-up this signal becomes an input to Strap SMBus Slave Address bit 2 • The EP SMBus slave is accessed with address[7:1] = 1110_XX0. • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired. • **DH89xxCL:** • This signal is used to Strap EP SMBus Slave Address bit 2 (High or Low). • The EP SMBus slave is accessed with address[7:1] = 1110_XX0 • Sampling occurs during the first 200 µs after assertion of EP_MAIN_PWR_OK

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 3 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE1_LED	EP_SMB_ADR3	O	LVTTL	R11	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK or EP_MAIN_PWR_OK See ***Note*** in the DH89xxCC Description column	1149.1	**DH89xxCC:** • Port 1 LED. Programmable LED, mode encoding set by GbE EEPROM. • On power-up this signal becomes an input to Strap SMBus Slave Address bit 3. • The EP SMBus slave is accessed with address[7:1] = 1110_XX0. • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired. • **DH89xxCL:** • This signal is used to Strap EP SMBus Slave Address bit 3 (High or Low) • The EP SMBus slave is accessed with address[7:1] = 1110_XX0 • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 4 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE2_LED	EP_VF_ENABLED	O	LVTTL	E4	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK or EP_MAIN_PWR_OK See ***Note*** in the DH89xxCC Description column	1149.1	**DH89xxCC:** • Port 2 LED. Programmable LED, mode encoding set by GbE EEPROM. • This signal can be use as a strapping function to enable or disable PCIe* SRIOV GBE2_LED: 0 = Disable SRIOV (External Pull-Down Required) 1 = Enable SRIOV (External Pull-up Required) • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired. • **DH89xxCL:** • This signal can be use as a strapping function to enable or disable PCIe* SRIOV 0 = Disable SRIOV (External Pull-Down Required) 1 = Enable SRIOV (Default) • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 5 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE3_LED	EP_RESET_SEQ	O	LVTTL	U9	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK or EP_MAIN_PWR_OK See ***Note*** in the DH89xxCC Description column	1149.1	**DH89xxCC:** • Port 3 LED. Programmable LED, mode encoding set by GbE EEPROM. • This signal can be use as a strapping function to control reset sequence. 0 = **Reserved** ***Note:*** • Should NOT be pulled LOW 1 = PCIe* CEM 2.0 Compliant Reset sequence (External Pull-up Required) • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as a LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired. • **DH89xxCL:** • This signal is a strap function that controls reset sequencing • This signal must be sampled as a 1 by allowing it to float or pulling it up 1 = PCIe* CEM 2.0 Compliant Reset sequence (Default) • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 6 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE0_SWDP0 GBE0_SWDP1	RSVD124 RSVD125	I/O	LVTTL	Y9 M8	2		1149.1	**DH89xxCC:** Software Defined Pins for function 0. • These pins are software programmable w/with input/output capability. These default to inputs upon power up, but may have their direction and output values defined in the EEPROM. The SDP bits may be mapped to the General Purpose Interrupt bits when configured as inputs. The GBE0_SWDP0 pin can be used as a watchdog output indication. All the SDP pins can be used as SFP sideband signals (TxDisable, present & TxFault). This product does not use these signals; it is available for SW control over SFP. • **DH89xxCL:** • Reserved • No Connect
GBE1_SWDP0 GBE1_SWDP1	RSVD126 RSVD127	I/O	LVTTL	R2 J3	2		1149.1	**DH89xxCC:** Software Defined Pins for function 1. • These pins are software programmable w/with input/output capability. These default to inputs upon power up, but may have their direction and output values defined in the EEPROM. The SDP bits may be mapped to the General Purpose Interrupt bits when configured as inputs. The GBE1_SDP0 pin can be used as a watchdog output indication. All the SDP pins can be used as SFP sideband signals (TxDisable, present & TxFault). This product does not use these signals; it is available for SW control over SFP. • **DH89xxCL:** • Reserved • No Connect

Table 32-8. LED, Software Defined, Miscellaneous Signals (Sheet 7 of 7)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCLDH89xxCL**							
GBE2_SWDP0 GBE2_SWDP1	RSVD128 RSVD129	I/O	LVTTL	R6 G4	2		1149.1	**DH89xxCC:** Software Defined Pins for function 2. • These pins are software programmable w/with input/output capability. These default to inputs upon power up, but may have their direction and output values defined in the EEPROM. The SDP bits may be mapped to the General Purpose Interrupt bits when configured as inputs. The GBE2_SDP0 pin can be used as a watchdog output indication. All the SDP pins can be used as SFP sideband signals (TxDisable, present & TxFault). This product does not use these signals; it is available for SW control over SFP. • **DH89xxCL:** • Reserved • No Connect
GBE3_SWDP0 GBE3_SWDP1	RSVD130 RSVD131	I/O	LVTTL	R4 H10	2		1149.1	**DH89xxCC:** Software Defined Pins for function 3. • These pins are software programmable w/with input/output capability. These default to inputs upon power up, but may have their direction and output values defined in the EEPROM. The SDP bits may be mapped to the General Purpose Interrupt bits when configured as inputs. The GBE3_SDP0 pin can be used as a watchdog output indication. All the SDP pins can be used as SFP sideband signals (TxDisable, present & TxFault). This product does not use these signals; it is available for SW control over SFP. • **DH89xxCL:** • Reserved • No Connect
TOTAL					16			

Notes:
1. For designs where GBE[3:0]_LED output signals are **NOT** used to drive platform GBE LEDs, the board level straps may be hardwired to the desired strap configuration.
2. See the *platform design collateral* for strap resistor values and implementation reference.

32.7 EndPoint Management SMBus Interface (Slave)

Table 32-9. EndPoint Management SMBus Interface Signals (Slave)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
EP_SMBCLK	I	LVTTL	B13	1	External pull-up required[1]	1149.1	Management SMBus Clock. One clock pulse is received for each data bit transferred. External pull-up resistor to VCCPEP3P3AUX is required. Resistor value should be calculated based on the bus load, see the platform design collateral.
EP_SMBDAT	I/OD	LVTTL	M12	1	External pull-up required	1149.1	Management SMBus Data. Stable during the high period of the clock (unless it is a start or stop condition). External pull-up resistor to VCCPEP3P3AUX is required. Resistor value should be calculated based on the bus load, see the platform design collateral.
EP_SMBALRT#	OD	LVTTL	C12	1	External pull-up required	1149.1	Management SMBus Alert. Acts as an interrupt of a slave device on SMBus. External pull-up resistor to VCCPEP3P3AUX is required. Resistor value should be calculated based on the bus load, see the platform design collateral.
TOTAL				3			

1. See the platform design collateral for resistor values.

32.8 PCI Express* Root Complex

Table 32-10. PCI Express* Root Complex Interface Signals [1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
PCIE_RC_RXp0 PCIE_RC_Rxn0	I	LV Diff	AG53 AG56	2		1149.1	PCI Express Root Complex Receive Data Pair Lane 0
PCIE_RC_RXp1 PCIE_RC_Rxn1	I	LV Diff	AJ53 AJ55	2		1149.1	PCI Express Root Complex Receive Data Pair Lane 1
PCIE_RC_RXp2 PCIE_RC_RXn2	I	LV Diff	AL56 AM55	2		1149.1	PCI Express Root Complex Receive Data Pair Lane 2
PCIE_RC_RXp3 PCIE_RC_RXn3	I	LV Diff	AN56 AN54	2		1149.1	PCI Express Root Complex Receive Data Pair Lane 3
PCIE_RC_TXp0 PCIE_RC_TXn0	O	LV Diff	AG49 AG50	2		1149.1	PCI Express Root Complex Transmit Data Pair Lane 0
PCIE_RC_TXp1 PCIE_RC_TXn1	O	LV Diff	AL47 AL49	2		1149.1	PCI Express Root Complex Transmit Data Pair Lane 1
PCIE_RC_TXp2 PCIE_RC_TXn2	O	LV Diff	AN46 AN47	2		1149.1	PCI Express Root Complex Transmit Data Pair Lane 2
PCIE_RC_TXp3 PCIE_RC_TXn3	O	LV Diff	AN50 AT50	2		1149.1	PCI Express Root Complex Transmit Data Pair Lane 3
TOTAL				16			

1. All transmit signals require AC coupling capacitors. See the PDG for capacitor values and routing guidelines.

32.9 Serial ATA Interface

The PCH supports only two SATA ports (ports 4 & 5).

Table 32-11. SATA Interface Signals (Sheet 1 of 2)[1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SATA4_RXN SATA4_RXP	I	LV Diff	AT57 AR56	2		1149.1	Serial ATA 4 Differential Receive Pair. These are inbound high-speed differential signals from Port 4. In compatible mode (IDE Mode), SATA Port 4 is the primary master of SATA Controller. ***Note:*** Port4 corresponds to signal SATA4
SATA4_TXN SATA4_TXP	O	LV Diff	AV49 AV47	2		1149.1	Serial ATA 4 Differential Transmit Pairs. These are outbound high-speed differential signals to Port 4. In compatible mode (IDE Mode), SATA Port 4 is the primary master of SATA Controller. ***Note:*** Port4 corresponds to signal SATA4
SATA5_GP / TEMP_ALERT# / GPIO49	I/O	LVTTL	BC46	1		1149.1	Serial ATA 5 General Purpose. This is an input pin which can be configured as an interlock switch corresponding to SATA Port 5. When used as an interlock switch status indication, this signal should be drive to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. Temperature Alert. Used as an alert (active low) to indicate to the external controller (such as, EC or SIO) that temperatures are out of range for the PCH or Memory Controller or the processor core. If interlock switches or Temp Alert are not required, this pin can be configured as GPIO Port 49.
SATA5_RXN SATA5_RXP	I	LV Diff	AU53 AT55	2		1149.1	Serial ATA 5 Differential Receive Pair. These are inbound high-speed differential signals from Port 5. In compatible mode (IDE Mode), SATA Port 5 is the secondary master of SATA Controller. ***Note:*** Port5 corresponds to signal SATA5
SATA5_TXN SATA5_TXP	O	LV Diff	BA50 BA49	2		1149.1	Serial ATA 5 Differential Transmit Pair. These are outbound high-speed differential signals to Port 5. In compatible mode (IDE Mode), SATA Port 5 is the secondary master of SATA Controller. ***Note:*** Port5 corresponds to signal SATA5
SDATAOUT0 / GPIO39	I/O	LVTTL	BR52	1		1149.1	SATA Serial GPIO Data Out 0. Driven by the controller to indicate the drive status in the following sequence: drive 4, 5...4,5... If SDATAOUT0 interface is not used, the signals can be used as GPIO Port 39.
SDATAOUT1 / GPIO48	I/O	LVTTL	BT51	1		1149.1	SATA Serial GPIO Data Out 1. Driven by the controller to indicate the drive status in the following sequence: drive 4, 5...4,5... If SDATAOUT1 interface is not used, the signals can be used as GPIO Port 48.

Table 32-11. SATA Interface Signals (Sheet 2 of 2)[1]

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SCLOCK / GPIO22	I/O	LVTTL	BR49	1		1149.1	SGPIO Reference Clock. The SATA controller uses rising edges of this clock to transmit serial data, and the target uses the falling edge of this clock to latch data. If SCLOCK interface is not used, this signal can be used as a GPIO Port 22.
SLOAD / GPIO38	I/O	LVTTL	BU49	1		1149.1	SATA Serial GPIO Load. The controller drives a '1' at the rising edge of SCLOCK to indicate either the start or end of a bit stream. A 4-bit vendor specific pattern will be transmitted right after the signal assertion. If SLOAD interface is not used, this signal can be used as a GPIO Port 38.
SATA_LED#	OD	LVTTL	BD38	1	External pull-up required[2]	1149.1	Serial ATA LED. This signal is an open-drain output pin driven during SATA command activity. It is to be connected to external circuitry that can provide the current to drive a platform LED. When active, the LED is on. When tri-stated, the LED is off. An external pull-up resistor to VCC3P3 is required.
SATA_VREF1P8	I	IO Supply	AT47	1			Serial ATA Reference Voltage. Connect to VCCA1P8_SATA, 1.8V Core reference voltage.
SATA4_GP / GPIO16	I/O	LVTTL	BC47	1		1149.1	Serial ATA 4 General Purpose. This is an input pin which can be configured as an interlock switch corresponding to SATA Port 4. When used as an interlock switch status indication, this signal should be drive to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. If interlock switches are not required, this pin can be configured as GPIO Port 16.
SATA_ICOMPO	I	Analog	AT44	1	External Pull-up 10.5KΩ 1%		Serial ATA Compensation Output. Connected pull-up resistor to VCCA1P8_SATA.
TOTAL				17			

1. All Transmit and Receive signals (SATA4/5_TXN/P and SATA4/5_RXN/P) require AC coupling capacitors. See the PDG for capacitor values and routing guidelines.
2. See the platform design collateral for resistor values.

32.10 LPC Interface

Note: BIOS booting is done via the SPI Boot Interface, NOT the LPC Interface.

Table 32-12. LPC Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
INIT3_3V#	O	LVTTL	BK50	1		1149.1	Initialization 3.3 V. INIT3_3V# is asserted driven out for 16 PCI clocks to reset the processor.
LAD[3:0]	I/O	LVTTL	BU39 BT41 BJ33 BR39	4	Internal pull-up	1149.1	Low Pin Count Address Data Bus. Data/Address signals for the LPC bus.
LDRQ0#	I	LVTTL	BJ36	1	Internal pull-up	1149.1	LPC Serial DMA/Master Request Input Bit 0. Used by LPC devices, such as Super I/O chips, to request DMA or bus master access. This signal is typically connected to external Super I/O device.
LDRQ1# / GPIO23	I/O	LVTTL	BG33	1		1149.1	LPC Serial DMA/Master Request Input Bit 1. Used by LPC devices, such as Super I/O chips, to request DMA or bus master access. This signal is typically connected to external Super I/O device. An internal pull-up resistor is provided on these signals. If LDRQ1# interface is not used, this signal can be used as a GPIO Port 23.
LFRAME#	O	LVTTL	BF33	1	Internal pull-up	1149.1	LPC Frame. This signal is multiplexed with the LPC LFRAME# signal.
TOTAL				8			

32.11 USB* Interface

Table 32-13. USB* Interface Signals (Sheet 1 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
OC0# / GPIO59 OC1# / GPIO40 OC2# / GPIO41 OC3# / GPIO42	I/O	LVTTL	BK12 BN4 BK10 BM18	4		1149.1	Overcurrent Indicators. These signals set corresponding bits in the USB* controllers to indicate that an overcurrent condition has occurred. OC[3:0]# may optionally be used as GPIO Ports [42,41,40,59]. ***Notes:*** 1. OC# pins are 3.3V and NOT 5 V tolerant. 2. OC# pins must be shared between ports. 3. OC#[3:0] can only be used for EHCI controller #1.
USB0_Dn USB0_Dp USB1_Dn USB1_Dp	I/O	LV Diff	BA2 BA5 BB3 BC2	4	Internal pull-down	1149.1	Universal Serial Bus Port [1:0] Differential. These differential pairs are used to transmit Data/Address/Command signals for ports 0 and 1. These ports can be routed to UHCI controller #1 or EHCI controller #1. ***Note:*** No external resistors are required on these signals. The driver contains an integrated 15 kΩ pull-down and provides an output driver impedance of 45 Ω which requires no external series resistor.
USB2_Dn USB2_Dp USB3_Dn USB3_Dp	I/O	LV Diff	BC4 BE4 BE2 BF3	4	Internal pull-down	1149.1	Universal Serial Bus Port [3:2] Differential. These differential pairs are used to transmit data/address/command signals for ports 2 and 3. These ports can be routed to UHCI controller #2 or the EHCI controller #1. ***Note:*** No external resistors are required on these signals. The driver contains an integrated 15 kΩ pull-down and provides an output driver impedance of 45 Ω which requires no external series resistor.
USB4_Dn USB4_Dp USB5_Dn USB5_Dp	I/O	LV Diff	BG2 BJ1 BF8 BF9	4	Internal pull-down	1149.1	Universal Serial Bus Port [5:4] Differential. These differential pairs are used to transmit Data/Address/Command signals for ports 4 and 5. These ports can be routed to UHCI controller #3 or the EHCI controller #1. ***Note:*** No external resistors are required on these signals. The driver contains an integrated 15 kΩ pull-down and provides an output driver impedance of 45 Ω which requires no external series resistor.

Table 32-13. USB* Interface Signals (Sheet 2 of 2)

Intel® Communications Chipset 89xx Series DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
USB_RBIASn	I	Analog	AU5	1	External pull-down required[1]		USB* Resistor Bias Complement. Analog connection point for an external resistor. Used to set transmit currents and internal load resistors.
USB_RBIASp	O	Analog	AU2	1	External pull-down required.		USB* Resistor Bias. Analog connection point for an external resistor. Used to set transmit currents and internal load resistors.
TOTAL				18			

1. See the platform design collateral for resistor values.

32.12 UART Interface

Table 32-14. UART Interface Signals (Sheet 1 of 3)

Intel® Communications Chipset 89xx Series DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SIU0_CTS#	I	LVTTL	BP33	1		1149.1	UART Port 0 Clear to Send: Active low, this pin indicates that data can be exchanged between UART port 0 and the external interface. This pin has no effect on the transmitter.
SIU0_DCD#	I	LVTTL	BG31	1		1149.1	UART Port 0 Data Carrier Detect: Active low, this pin indicates that data carrier has been detected by the external agent for UART port 0.
SIU0_DSR#	I	LVTTL	BU32	1		1149.1	UART Port 0 Data Set Ready: Active low, this pin indicates that the external agent is ready to communicate with UART port 0. This pin has no effect on the transmitter.

Table 32-14. UART Interface Signals (Sheet 2 of 3)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SIU0_DTR#	O	LVTTL	BP43	1	Weak Internal pull-up	1149.1	UART Port 0 Data Terminal Ready: When low, this pin informs the modem or data set that the UART port 0 is ready to establish a communication link. SIW Configuration Port Address Select Strap: this strap selects the IO address for the SIW configuration port. Sampling occurs on the rising edge of PWROK. The straps are defined as follows: 0 = IO Addresses 02Eh and 02Fh (Pull-down required) 1 = IO Addresses 04Eh and 04Fh (default)
SIU0_RI#	I	LVTTL	BK33	1		1149.1	UART Port 0 Ring Indicator: Active low, this pin indicates that a telephone ringing signal has been received by the external agent for UART port 0. ***Note:*** This pin is Modem Status Input whose condition can be tested by the processor by reading bit 6 (RI) of the MSR. Bit 6 is the complement of the RI# signal. Bit 2 (TERI) of the MSR indicates whether the RI# input has transitioned back to an inactive state. When the RI bit of the MSR changes from a 1 to 0 an interrupt is generated if the Modem Status Interrupt is enabled.
SIU0_RTS#	O	LVTTL	BM33	1		1149.1	UART Port 0 Request To Send: When low this pin informs the modem or data set that UART port 0 is wants to send data on an established communication link. The RTS# output signal can be set to an active low by programming the RTS (bit 1) of the Modem Control Register to a logic '1'. A Reset operation sets this signal to its inactive state (logic '1'). LOOP mode operation holds this signal in its inactive state.
SIU0_RXD	I	LVTTL	BJ31	1		1149.1	UART Port 0 Serial Data Input: Serial data input form device pin to the receive port for UART port 0.
SIU0_TXD	O	LVTTL	BT43	1		1149.1	UART Port 0 Serial Data Output: Serial data output to the communication peripheral/modem or data set for UART port 0. Upon reset, the TXD pins will be set to MARKING condition (logic '1' state).
SIU1_CTS#	I	LVTTL	BT35	1		1149.1	UART Port 1 Clear to Send: Active low, this pin indicates that data can be exchanged between UART port 1 and the external interface. This pin has no effect on the transmitter.

Table 32-14. UART Interface Signals (Sheet 3 of 3)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SIU1_DCD#	I	LVTTL	BK36	1		1149.1	UART Port 1 Data Carrier Detect: Active low, this pin indicates that data carrier has been detected by the external agent for UART port 1.
SIU1_DSR#	I	LVTTL	BP35	1		1149.1	UART Port 1 Data Set Ready: Active low, this pin indicates that the external agent is ready to communicate with UART port 1. This pin has no effect on the transmitter.
SIU1_DTR#	O	LVTTL	BN52	1		1149.1	UART Port 1 Data Terminal Ready: When low, this pin informs the modem or data set that the UART port 1 is ready to establish a communication link.
SIU1_RI#	I	LVTTL	BN37	1		1149.1	UART Port 1 Ring Indicator: Active low, this pin indicates that a telephone ringing signal has been received by the external agent for UART port 1. ***Note:*** This pin is Modem Status Input whose condition can be tested by the processor by reading bit 6 (RI) of the MSR. Bit 6 is the complement of the RI# signal. Bit 2 (TERI) of the MSR indicates whether the RI# input has transitioned back to an inactive state. When the RI bit of the MSR changes from a 1 to 0 an interrupt is generated if the Modem Status Interrupt is enabled.
SIU1_RTS#	O	LVTTL	BU36	1		1149.1	UART Port 1 Request To Send: When low this pin informs the modem or data set that UART port 1 is wants to send data on an established communication link. The RTS# output signal can be set to an active low by programming the RTS (bit 1) of the Modem Control Register to a logic '1'. A Reset operation sets this signal to its inactive state (logic '1'). LOOP mode operation holds this signal in its inactive state.
SIU1_RXD	I	LVTTL	BM36	1		1149.1	UART Port 1 Serial Data Input: Serial data input form device pin to the receive port for UART port 1.
SIU1_TXD	O	LVTTL	BK41	1		1149.1	UART Port 1 Serial Data Output: Serial data output to the communication peripheral/modem or data set for UART port 1. Upon reset, the TXD pins will be set to MARKING condition (logic '1' state).
WDT_TOUT#	O	LVTTL	BT37	1		1149.1	The WatchDog Timer output. The signal is driven low when the main 35-bit down counter reaches zero during the second stage. For use by board management controller.
TOTAL				17			

32.13 Host SMBus (Master) Interface

Table 32-15. Host SMBus (Master) Interface Signals (Sheet 1 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
MST_SMBCLK	I/OD	LVTTL	BL2	1	External pull-up required[1]	1149.1	Master SMBus Clock. One clock pulse is generated for each data bit transferred. External pull-up resistor to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral.
MST_SMBDAT	I/OD	LVTTL	BL4	1	External pull-up required.	1149.1	Master SMBus Data. Stable during the high period of the clock (unless it is a start or stop condition). External pull-up resistor to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral.
MST_SMBALERT# / GPIO11	I/O	LVTTL	BG15	1	When used as SMBUS: External pull-up required.	1149.1	SMBus Alert. This signal is used to wake the system or generate SMI#. External pull-up resistor to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral. This signal can also be configured to GPIO Port 11.
SML1CLK / GPIO58	I/O	LVTTL	BJ20	1	When used as SMBUS: External pull-up required.	1149.1	System Management Link 1 Clock. SMBus link to external BMC. External pull-up resistor to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1CLK interface is not used, the signals can be used as GPIO Port 58.

Table 32-15. Host SMBus (Master) Interface Signals (Sheet 2 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SML1DAT / GPIO75	I/O	LVTTL	BK20	1	When used as SMBUS: External pull-up required.	1149.1	System Management Link 1 Data. SMBus link to external BMC. External pull-up required to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1DAT interface is not used, the signals can be used as GPIO Port 75.
SML1ALERT# / GPIO74	I/O	LVTTL	BU12	1	When used as SMBUS: External pull-up required.	1149.1	System Management Link Alert 1. This signal can be connected to an external BMC. External pull-up resistor to VCCSUS3P3 is required. Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1ALERT# interface is not used, the signals can be used as GPIO Port 74.
TOTAL				6			

1. See the platform design collateral for resistor values.

32.14 Serial Peripheral Interface Boot Interface

Note: Use SPI Flash to boot the system.

Table 32-16. SPI Boot Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SPI_CLK	O	LVTTL	BD41	1		1149.1	SPI Clock. SPI clock signal, during idle the bus owner will drive the clock signal low.
SPI_CS0#	O	LVTTL	BJ55	1		1149.1	SPI Chip Select 0. Used as the SPI bus request signal.
SPI_CS1#	O	LVTTL	BC56	1	Weak Internal pull-up	1149.1	SPI Chip Select 1. Used as the SPI bus request signal.
SPI_MISO	I	LVTTL	BH50	1	Weak Internal pull-up	1149.1	SPI Master IN Slave OUT. Data input pin.
SPI_MOSI	O	LVTTL	BC54	1		1149.1	SPI Master OUT Slave IN. Data output pin.
TOTAL				5			

32.15 Interrupt Interface

Table 32-17. Interrupt Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SERIRQ	IO	LVTTL	BL56	1	External pull-up required[1]	1149.1	Serial Interrupt Request. This pin implements the serial interrupt protocol.
TOTAL				1			

1. See the platform design collateral for resistor values.

32.16 Processor Interface

Table 32-18. Processor Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
A20GATE	I	LVTTL	BL54	1	Depends on driver	1149.1	A20 Gate. A20GATE is from the keyboard controller. The signal acts as an alternative method to force the A20M# signal active. It saves the external OR gate needed with various other chipsets.
CPUPWRGD	O	LVCMOS	BB57	1		1149.1	CPU Power Good. This signal should be connected to the processor's PWRGOOD input to indicate when the processor power is valid. Logic level is determined by the voltage value of VCCPCPU.
RCIN#	I	LVTTL	BF47	1	Depends on driver	1149.1	Keyboard Controller Reset CPU. The keyboard controller can generate INIT# to the processor. This saves the external OR gate with the PCH's other sources of INIT#. When the PCH detects the assertion of this signal, INIT# is generated for 16 PCI clocks. ***Note:*** The PCH will ignore RCIN# assertion during transitions to the S1, S3, S4, and S5 states.
TOTAL				3			

32.17 Power Management Interface

Table 32-19. Power Management Interface Signals (Sheet 1 of 3)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DRAMPWRGD	OD	LVTTL	BT17	1	External pull-up required[1]	1149.1	DRAM Power OK. This signal is connected to the processor's DRAM_PWR_OK_Cxx pins. The PCH asserts this pin to indicate when DRAM power is on. ***Note:*** • This is an open drain output and requires an external pull-up. • Refer to the platform design collateral for design implementation recommendation.
MEPWROK	I	LVTTL	BR16	1			Management Engine Power OK. When asserted, indicates that power to the ME subsystem is stable. Connect to PWROK on platform.
PECI	I/O	SSTL	BB55	1		1149.1	Platform Environment Control Interface: Single-wire, serial bus. Connect to corresponding pin of the processor for accessing processor digital thermometer. Logic level is determined by the value of VCCPCPU.
PLTRST#	O	LVTTL	BG12	1		1149.1	Platform Reset. The PCH asserts PLTRST# to reset devices on the platform (for example, SIO, LAN, Processor, etc.). The PCH asserts PLTRST# during power-up and when S/W initiates a hard reset sequence through the Reset Control register (I/O Register CF9h). The PCH drives PLTRST# inactive a minimum of 1 ms after both PWROK and SYS_PWROK are driven high. The PCH drives PLTRST# active a minimum of 1 ms when initiated through the Reset Control register (I/O Register CF9h). ***Note:*** PLTRST# is in the VCCSUS3P3 well.
PMSYNC	O	LVCMOS	BA53	1		1149.1	Power Management Sync. Provides state information from the PCH to the Processor relevant to C-state transitions. Logic level is determined by the value of VCCPCPU.
PWRBTN#	I	LVTTL	BG20	1	Internal pull-up	1149.1	Power Button. The Power Button will cause SMI# or SCI to indicate a system request to go to a sleep state. If the system is already in a sleep state, this signal will cause a wake event. If PWRBTN# is pressed for more than 4 seconds, this will cause an unconditional transition (power button override) to the S5 state. Override will occur even if the system is in the S1-S4 states. This signal has an internal pull-up resistor and has an internal 16 ms de-bounce on the input.

Table 32-19. Power Management Interface Signals (Sheet 2 of 3)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
RI#	I	LVTTL	BK17	1	Internal pull-up	1149.1	Ring Indicate. This signal is an input from a modem. It can be enabled as a wake event, and this is preserved across power failures.
SLP_S3#	O	LVTTL	BN19	1		1149.1	S3 Sleep Control. SLP_S3# is for power plane control. This signal shuts off power to all non-critical systems when in S3 (Suspend To RAM), S4 (Suspend to Disk), or S5 (Soft Off) states.
SLP_S4#	O	LVTTL	BN17	1		1149.1	S4 Sleep Control. SLP_S4# is for power plane control. This signal shuts power to all non-critical systems when in the S4 (Suspend to Disk) or S5 (Soft Off) state. ***Note:*** This pin must be used to control the DRAM power in order to use the PCH's DRAM power-cycling feature.
SLP_S5# / GPIO63	I/O	LVTTL	BF20	1		1149.1	S5 Sleep Control. SLP_S5# is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) states. If SLP_S5# interface is not used, the signals can be used as GPIO Port 63.
SUS_CLK / GPIO62	I/O	LVTTL	BP13	1		1149.1	Suspend Clock. This clock is an output of the RTC generator circuit. It is used by other chips for refresh clock. If SUS_CLK interface is not used, the signals can be used as GPIO Port 62.
SUS_STAT# / GPIO61	I/O	LVTTL	BM13	1		1149.1	Suspend Status: This signal is asserted by the PCH to indicate that the system will be entering a low power state soon. This can be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It can also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes. If SUS_STAT# interface is not used, this signal can be used as a GPIO Port 61.

Table 32-19. Power Management Interface Signals (Sheet 3 of 3)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
SYS_PWROK	I	LVTTL	BP15	1			System Power OK. This generic power good input to the PCH is driven and utilized in a platform-specific manner. While PWROK always indicates that the CORE well of the PCH is stable, SYS_PWROK is used to inform the PCH that power is stable to some other system component(s) and the system is ready to start the exit from reset. The particular component(s) associated with SYS_PWROK can vary across platform types supported by the same generation of the PCH. Depending on the platform, the PCH may expect (and wait) for SYS_PWROK at different stages of the boot flow before continuing. ***Note:*** Consult the platform design collateral for instructions on how to connect SYS_PWROK for that platform.
SYS_RESET#	I	LVTTL	BJ48	1	External pull-up required.	1149.1	System Reset. This pin forces an internal reset after being debounced. The PCH will reset immediately if the SMBus is idle; otherwise, it will wait up to 25 ms ± 2 ms for the SMBus to idle before forcing a reset on the system.
THRMTRIP#	I	LVCMOS	BF52	1	External pull-up required.	1149.1	Thermal Trip. When low, this signal indicates that a thermal trip from the processor occurred, and the PCH will immediately transition to a S5 state. The PCH will not wait for the processor stop grant cycle since the processor has overheated. Logic level is determined by the value of VCCPCPU.
WAKE#	I	LVTTL	BU19	1		1149.1	PCI Express Wake Event. Sideband wake signal on PCI Express asserted by components requesting wake up.
TOTAL				16			

1. See the platform design collateral for resistor values.

32.18 Thermal Sensor Current Reference

Table 32-20. Thermal Sensor Current Reference

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
TS0_IREF_N	I	Analog	D53	1	External pull-down 8.06K 1% to VSS		Thermal Sensor 0 Current Reference. Connect to an external pull down resistor to ground.
TS1_IREF_N	I	Analog	BH9	1	External pull-down 8.06K 1% to VSS		Thermal Sensor 1 Current Reference. Connect to an external pull down resistor to ground.
TOTAL				2			

32.19 Miscellaneous Interface

Table 32-21. Miscellaneous Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
BMBUSY# / GPIO0	I/O	LVTTL	BT45	1		1149.1	Bus Master Busy. This signal is used to support the C3 state. It indicates that a bus master device is busy. When this signal is asserted, the BM_STS bit will be set. If this signal goes active in a C3 state, it is treated as a break event. ***Note:*** This signal is internally synchronized using the PCICLK and a two-stage synchronizer. It does not need to meet any particular setup or hold time. This signal can also be used as GPIO Port 0.
EP_CRU_EN	I	LVTTL	L12	1		1149.1	EndPoint CRU (Clock Resource Unit) Enable. Enables CRU Clock inputs CRU_CLK100[P/N]
TOTAL				2			

32.20 General Purpose Input/Output Interface

There are 68 GPIO pins (multiplexed or dedicated).

Table 32-22. General Purpose I/O Interface Signals (Sheet 1 of 9)

Intel® Communications Chipset 89xx Series DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
BMBUSY#/GPIO0	I/O	BMBUSY#	BT45	1	I	CORE		**Bus Master Busy.** • This signal is used to support the C3 state. It indicates that a bus master device is busy. When this signal is asserted, the BM_STS bit will be set. If this signal goes active in a C3 state, it is treated as a break event. ***Note:*** This signal is internally synchronized using the PCICLK and a two-stage synchronizer. It does not need to meet any particular setup or hold time. This signal can also be used as GPIO Port 0.
GPIO1	I/O	GPI	BN47	1	I	CORE		General Purpose I/O Port 1.
GPIO2 [1]	I/O	GPI	BK43	1	I	CORE		General Purpose I/O Port 2.
GPIO3	I/O	GPI	BR42	1	I	CORE		General Purpose I/O Port 3.
GPIO4	I/O	GPI	BJ43	1	I	CORE		General Purpose I/O Port 4.
GPIO5	I/O	GPI	BM45	1	I	CORE		General Purpose I/O Port 5.
GPIO6	I/O	GPI	BP53	1	I	CORE		General Purpose I/O Port 6.
GPIO7	I/O	GPI	BT47	1	I	CORE		General Purpose I/O Port 7.
GPIO8	I/O	GPO	BU16	1	O (High)	SUS	Weak Internal pull-up for strap.	General Purpose I/O Port 8. • This signal has a weak internal pull-up and must not be pulled low during boot up.
GPIO9[2]	I/O	Native	BN21	1	I	SUS		General Purpose I/O Port 9[3]
GPIO10	I/O	Native	BG22	1	I	SUS		General Purpose I/O Port 10.
MST_SMBALERT#/ GPIO11	I/O	MST_SMBALERT#	BG15	1	I	SUS	External pull-up required	**Host SMBus Alert.** • This signal is used to wake the system or generate SMI#. • External pull-up resistor to VCCSUS3P3 is required. • Resistor value should be calculated based on the bus load, (See the platform design collateral for Resistor value) This signal can also be configured to GPIO Port 11.
GPIO12	I/O	Native	BF15	1	O (Low)	SUS		General Purpose I/O Port 12.

Table 32-22. General Purpose I/O Interface Signals (Sheet 2 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
GPIO13[4]	I/O	GPI	BJ25	1	I	SUS	Weak internal pull-down	General Purpose I/O Port 13.
GPIO14	I/O	Native	BP23	1	I	SUS		General Purpose I/O Port 14.
GPIO15	I/O	GPO	BJ10	1	O (Low)	SUS		General Purpose I/O Port 15.
SATA4_GP/GPIO16	I/O	GPI	BC47	1	I	CORE		**Serial ATA 4 General Purpose.** • This is an input pin which can be configured as an interlock switch corresponding to SATA Port 4. When used as an interlock switch status indication, this signal should be drive to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. If interlock switches are not required, this pin can be configured as GPIO Port 16.
GPIO17	I/O	GPI	BF38	1	I	CORE	Platform Dependent	**DMI Voltage Strap.** • This signal strapping sets the DMI termination voltage. • See PDG for additional information. This signal can be used as a GPIO Port 17
GPIO18[5]	I/O	Native	BG56	1	I	CORE		General Purpose I/O Port 18.
GPIO19	I/O	GPI	BF43	1	I	CORE		General Purpose I/O Port 19.
GPIO20	I/O	Native	BC50	1	I	CORE		General Purpose I/O Port 20.
GPIO21	I/O	GPI	BF49	1	I	CORE		General Purpose I/O Port 21.
SCLOCK/GPIO22	I/O	GPI	BR49	1	I	CORE		**SGPIO Reference Clock.** • The SATA controller uses rising edges of this clock to transmit serial data, and the target uses the falling edge of this clock to latch data. If SCLOCK interface is not used, this signal can be used as a GPIO Port 22.

Table 32-22. General Purpose I/O Interface Signals (Sheet 3 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
LDRQ1#/GPIO23	I/O	LDRQ1#	BG33	1	I	CORE		**LPC Serial DMA/Master Request Input Bit 1**. • Used by LPC devices, such as Super I/O chips, to request DMA or bus master access. This signal is typically connected to external Super I/O device. An internal pull-up resistor is provided on these signals. If LDRQ1# interface is not used, this signal can be used as a GPIO Port 23.
GPIO24	I/O	GPO	BP7	1	O (Low)	SUS		General Purpose I/O Port 24.
GPIO25	I/O	Native	BK25	1	I	SUS		General Purpose I/O Port 25.
GPIO26	I/O	Native	BM26	1	I	SUS		General Purpose I/O Port 26.
GPIO27	I/O	GPO	BR6	1	O (Low)	SUS		General Purpose I/O Port 27.
GPIO28	I/O	GPI	BG17	1	I	SUS		General Purpose I/O Port 28.
GPIO30	I/O	GPI	BM10	1	I	SUS	Internal pull-down	General Purpose I/O Port 30.
GPIO31	I/O	GPI	BK15	1	I	SUS		General Purpose I/O Port 31.
GPIO32	I/O	Native	BG38	1	O (High)	CORE		General Purpose I/O Port 32.
GPIO33	I/O	GPO	BN41	1	O (High)	CORE	Weak Internal pull-up	**Flash Descriptor Security Overwrite**. • This signal is used to set the security override strap on the PCH. If sampled low, the Flash Descriptor Security will be overridden. If high, the security measures defined in the Flash Descriptor will be in effect. This strap should only be enabled (pulled low) in manufacturing environments using an external pull-down resistor. **GPIO33** 0 = Enable (Pull-Down Required) 1 = Disable (Default) • When the Security Overwrite is enable, it allows permission to every master to read and write to the entire Flash Components including areas outside the defined regions.
GPIO34	I/O	GPI	BN49	1	I	CORE		General Purpose I/O Port 34.
GPIO35	I/O	GPO	BF46	1	O (Low)	CORE		General Purpose I/O Port 35.
GPIO36	I/O	GPI	BN54	1	I	CORE		General Purpose I/O Port 36.

Table 32-22. General Purpose I/O Interface Signals (Sheet 4 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
ADR/GPIO37	I/O	GPI	BG53	1	I	CORE		General Purpose I/O Port 37. • Can be used for ADR (Asynchronous DRAM Refresh) trigger on platform. Only supported if processor supports ADR. • HARDWARE activation mechanism that triggers memory controller of CPU to put SDRAM into self-refresh mode. Activation of ADR flushes contents of some write data buffers to the DIMM before self refresh entry.
SLOAD/GPIO38	I/O	GPI	BU49	1	I	CORE		**SATA Serial GPIO Load**. • The controller drives a '1' at the rising edge of SCLOCK to indicate either the start or end of a bit stream. A 4-bit vendor specific pattern will be transmitted right after the signal assertion. If SLOAD interface is not used, this signal can be used as a GPIO Port 38.
SDATAOUT0/GPIO39	I/O	GPI	BR52	1	I	CORE		**SATA Serial GPIO Data Out 0.** • Driven by the controller to indicate the drive status in the following sequence: drive 4, 5...4,5... If SDATAOUT0 interface is not used, the signals can be used as GPIO Port 39.
OC1#/GPIO40	I/O	OC1#	BN4	1	I	SUS		**Overcurrent Indicators.** • These signals set corresponding bits in the USB* controllers to indicate that an overcurrent condition has occurred. • OC[3:0]# may optionally be used as GPIO Ports [42,41,40,59]. ***Notes:*** 1. OC# pins are 3.3V and NOT 5 V tolerant. 2. OC# pins must be shared between ports 3. OC#[3:0] can only be used for EHCI controller #1

Table 32-22. General Purpose I/O Interface Signals (Sheet 5 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
OC2#/GPIO41	I/O	OC2#	BK10	1	I	SUS		**Overcurrent Indicators.** • These signals set corresponding bits in the USB* controllers to indicate that an overcurrent condition has occurred. • OC[3:0]# may optionally be used as GPIO Ports [42,41,40,59]. ***Notes:*** 1. OC# pins are 3.3V and NOT 5 V tolerant. 2. OC# pins must be shared between ports 3. OC#[3:0] can only be used for EHCI controller #1
OC3#/GPIO42	I/O	OC3#	BM18	1	I	SUS		**Overcurrent Indicators.** • These signals set corresponding bits in the USB* controllers to indicate that an overcurrent condition has occurred. • OC[3:0]# may optionally be used as GPIO Ports [42,41,40,59]. ***Notes:*** 1. OC# pins are 3.3V and NOT 5 V tolerant. 2. OC# pins must be shared between ports 3. OC#[3:0] can only be used for EHCI controller #1
GPIO43	I/O	Native	BK22	1	I	SUS		General Purpose I/O Port 43.
GPIO44	I/O	Native	BT23	1	I	SUS		General Purpose I/O Port 44.
GPIO45	I/O	Native	BT21	1	I	SUS		General Purpose I/O Port 45.
GPIO46	I/O	Native	BM20	1	I	SUS		General Purpose I/O Port 46.
GPIO47	I/O	Native	BU9	1	I	SUS		General Purpose I/O Port 47.
SDATAOUT1/ GPIO48	I/O	GPI	BT51	1	I	CORE		**SATA Serial GPIO Data Out 1**. • Driven by the controller to indicate the drive status in the following sequence: drive 4, 5...4,5... If SDATAOUT1 interface is not used, the signals can be used as GPIO Port 48.

Table 32-22. General Purpose I/O Interface Signals (Sheet 6 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
SATA5_GP/ TEMP_ALERT#/ GPIO49	I/O	GPI	BC46	1	I	CORE		**Serial ATA 5 General Purpose.** • This is an input pin which can be configured as an interlock switch corresponding to SATA Port 5. When used as an interlock switch status indication, this signal should be drive to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open. **Temperature Alert.** • Used as an alert (active low) to indicate to the external controller (such as, EC or SIO) that temperatures are out of range for the PCH or Memory Controller or the processor core. If interlock switches or Temp Alert are not required, this pin can be configured as GPIO Port 49.
GPIO50	I/O	Native	BG43	1	I	CORE		General Purpose I/O Port 50.
BBS1/GPIO51	I/O	BBS1	BM38	1	O (High)	CORE	Weak Internal pull-up	**BIOS Boot Strap 1**. BBS1 BBS0 0 0 NOT VALID 0 1 NOT VALID 1 0 NOT VALID 1 1 SPI BOOT If BBS1 interface is not used, the signals can be used as GPIO Port 51.
GPIO52	I/O	Native	BF36	1	I	CORE		General Purpose I/O Port 52. Not Multiplexed.
GPIO53	I/O	Native	BM42	1	O (High)	CORE	Platform Dependent. Weak internal pull-up	**DMI Coupling Strap.** 0 = AC Coupling (Pull-Down Required) 1 = DC Coupling (Default) • See PDG for additional information.
GPIO54	I/O	Native	BR46	1	I	CORE		General Purpose I/O Port 54.

Table 32-22. General Purpose I/O Interface Signals (Sheet 7 of 9)

Intel® Communications Chipset 89xx Series DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
GPIO55	I/O	Native	BP45	1	O (High)	CORE	Weak Internal pull-up	**BIOS Boot-Block Swap** • This mode allows the PCH to swap the Top-Block in the SPI (the boot block) with another location. **GPIO55** 0 = Enable Top-Block Swap (Pull-down Required) 1 = Disable Top-Block Swap (Default) ***Note:*** The internal pull-up is disabled after PLTRST# de-asserts. If the signal is sampled low, this indicates that the system is strapped to the "Top-Block Swap" mode (the PCH inverts A16 for all cycles targeting BIOS space). • The status of this strap is readable via the Top Swap bit (Chipset Config Registers:Offset 3414h:bit 0). Software will not be able to clear the Top-Swap bit until the system is rebooted without GPIO55 being pulled down.
GPIO56	I/O	Native	BR9	1	I	SUS		General Purpose I/O Port 56.
GPIO57	I/O	GPI	BN11	1	I	SUS		General Purpose I/O Port 57.
SML1CLK/GPIO58	I/O	SML1CLK	BJ20	1	I	SUS	When used as SMBUS: External pull-up required[6]	**System Management Link 1 Clock:** • SMBus link to external BMC. • External pull-up resistor to VCCSUS3P3 is required. • Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1CLK interface is not used, the signals can be used as GPIO Port 58.

Table 32-22. General Purpose I/O Interface Signals (Sheet 8 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
OC0#/GPIO59	I/O	OC0#	BK12	1	I	SUS		**Overcurrent Indicators.** • These signals set corresponding bits in the USB* controllers to indicate that an overcurrent condition has occurred. • OC[3:0]# may optionally be used as GPIO Ports [42,41,40,59]. ***Notes:*** 1. OC# pins are 3.3V and NOT 5 V tolerant. 2. OC# pins must be shared between ports 3. OC#[3:0] can only be used for EHCI controller #1
GPIO60	I/O	Native	BR12	1	I	SUS		General Purpose I/O Port 60.
SUS_STAT#/ GPIO61	I/O	SUS_STAT#	BM13	1	O (High)	SUS		**Suspend Status**: • This signal is asserted by the PCH to indicate that the system will be entering a low power state soon. This can be monitored by devices with memory that need to switch from normal refresh to suspend refresh mode. It can also be used by other peripherals as an indication that they should isolate their outputs that may be going to powered-off planes. If SUS_STAT# interface is not used, this signal can be used as a GPIO Port 61.
SUS_CLK/GPIO62	I/O	SUS_CLK	BP13	1	O (Low)	SUS		**Suspend Clock.** • This clock is an output of the RTC generator circuit. It is used by other chips for refresh clock. If SUS_CLK interface is not used, the signals can be used as GPIO Port 62.
SLP_S5#/GPIO63	I/O	SLP_S5#	BF20	1	O (High)	SUS		**S5 Sleep Control.** • SLP_S5# is for power plane control. • This signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) states. If SLP_S5# interface is not used, the signals can be used as GPIO Port 63.
GPIO72	I/O	Native	BT13	1	I	SUS		General Purpose I/O Port 72.

Table 32-22. General Purpose I/O Interface Signals (Sheet 9 of 9)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Default Mode**	**Ball Map**	**Ball Count**	**Default Direction and Logic State**	**Power Well**	**Internal/External Resistor Pull-Up/Down**	**Description**
GPIO73	I/O	Native	BR22	1	I	SUS		General Purpose I/O Port 73.
SML1ALERT#/ GPIO74	I/O	SML1ALERT#	BU12	1	I	SUS	When used as SMBUS: External pull-up required.	**System Management Link 1 Alert.** • This signal can be connected to an external BMC. • External pull-up resistor to VCCSUS3P3 is required. • Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1ALERT# interface is not used, the signals can be used as GPIO Port 74.
SML1DAT/GPIO75	I/O	SML1DAT	BK20	1	I	SUS	When used as SMBUS: External pull-up required.	**System Management Link 1 Data.** • SMBus link to external BMC. • External pull-up required to VCCSUS3P3 is required. • Resistor value should be calculated based on the bus load, see the platform design collateral. If SML1DAT interface is not used, the signals can be used as GPIO Port 75.
TOTAL				67				

1. When this signal is configured as GPO, the output stage is an open drain.
2. When the multiplexed GPIO is used as GPIO functionality, care should be taken to ensure the signal is stable in its inactive state of the native functionality, immediately after reset until it is initialized to GPIO functionality.
   Multiplexed signals is visible or Intel reserved.
3. For GPIOs where Native Mode is configured using SPI Soft Strap, the corresponding GPIO_USE_SEL bits for these GPIOs have no effect. The GPIO_USE_SEL bits for these GPIOs may change to reflect the Soft-Strap configuration even though GPIO Lockdown Enable (GLE) bit is set.
4. The functionality that is multiplexed with the GPIO may not be used in desktop configuration.
5. GPIO18 is configured as an input in default mode. GPIO18 will toggle at a frequency of approximately 1 Hz when the signal is programmed as an output (via GP_IO_SEL Register) by BIOS or system configuration.
6. See the platform design collateral for resistor values.

Note: GPIO[29], GPIO[71:64], and GPIO[95:76] do not exit or are reserved and cannot be used.

32.21 Real Time Clock (RTC) Interface

Table 32-23. Real Time Clock (RTC) Interface Signals (Sheet 1 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
RTCX1	I	Analog	BR26	1			Crystal Input 1. This signal is connected to the 32.768 kHz crystal. If no external crystal is used, then RTCX1 can be driven with the desired clock rate.
RTCX2	O	Analog	BU26	1			Crystal Input 2. This signal is connected to the 32.768 kHz crystal. If no external crystal is used, then RTCX2 should be left floating.
RTCRST#	I	LVTTL	BJ27	1	External RC Circuit required[1]		RTC Reset. When asserted, this signal resets register bits in the RTC well. ***Notes:*** 1. Unless CMOS is being cleared (only to be done in the G3 power state), the RTCRST# input must always be high when all other RTC power planes are on. 2. In the case where the RTC battery is not working or missing on the platform, the RTCRST# pin must rise before the RSMRST# pin. 3. Requires an external RC circuit. see Reference Board Schematic or PDG.
SRTCRST#	I	LVTTL	BK31	1	External RC Circuit required.		Secondary RTC Reset: This signal resets the manageability register bits in the RTC well when the RTC battery is removed. ***Notes:*** 1. The SRTCRST# input must always be high when all other RTC power planes are on. 2. In the case where the RTC battery is not working or missing on the platform, the SRTCRST# pin must rise before the RSMRST# pin. See Reference board schematic or PDG
INTRUDER#	I	LVTTL	BM25	1	External pull-up 1MΩ		Intruder Detect. This signal can be set to disable the system, when the system enclosure is open. This signal's status is readable, so it can be used like a GPI if the Intruder Detection is not needed. This pin requires an external pull-up pin of 1MΩ to VCC3P3_RTC.
IVCC_RTC	I	Analog	BP25	1	External Cap to VSS required		Internal RTC Voltage Regulator. Generates internally a 1.5V voltage regulator, which requires an external 0.1uF decoupling capacitor.

Table 32-23. Real Time Clock (RTC) Interface Signals (Sheet 2 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
PWROK	I	LVTTL	BK27	1			Power OK. When asserted, PWROK is an indication to the PCH that all of its core power rails have been stable for 10 ms. PWROK can be driven asynchronously. When PWROK is negated, the PCH asserts PLTRST#. ***Note:*** It is required that the power rails associated with PCIe* typically the 3.3V, 5V, and 12V core well rails) have been valid for 99ms prior to PWROK assertion in order to comply with the 100ms PCIe* 1.1 specification on PLTRST# deassertion. PWROK must not glitch, even if RSMRST# is low.
RSMRST#	I	LVTTL	BN27	1	External pull-down required.		Resume Well Reset. This signal is used for resetting the resume power plane logic. This signal must be asserted for at least 10 ms after the suspend power wells are valid. When deasserted, this signal is an indication that the suspend power wells are stable.
TOTAL				8			

1. See the platform design collateral for resistor and capacitor values.

32.22 System Input Clock

Table 32-24. Clock Input Interface Signals (Sheet 1 of 2)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		I/O Type	Technology	Ball Map	Ball Count	Internal/ External Resistor Pull-Up/ Down	BScan Support	Description
DH89xxCC	DH89xxCL							
CRU_CLK100N CRU_CLK100P		I	LV Diff	J22 J25	2			Clock and Reset Unit 100 MHz Differential input Clock.
DMI_CLK100N DMI_CLK100P		I	LV Diff	AE56 AE54	2			Direct Media Interface 100 MHz Differential Clock. These signals are used to run the Direct Media Interface. Runs at 100 MHz.
GBE_CLK100N GBE_CLK100P	RSVD108 RSVD107	I	LV Diff	AF3 AF1	2			**DH89xxCL:** • Reserved • No Connect. **DH89xxCC:** GbE 100 MHz Differential Clock with 50 ppm max jitter. • External SerDes/SGMII differential 100MHz reference clock from an external generator. This clock must be running from the auxiliary power. • GBE_AUX_PWR_OK is asserted only when this clock is stable.
PCICLK		I	LVTTL	BJ38	1		1149.1	LPC clock. PCI clock used for the LPC bus (up to 33 MHz).
PCIE_EP_CLK100N PCIE_EP_CLK100P		I	LV Diff	U43 R41	2			PCI Express End Point Clock (Differential): A 100 MHz differential clock input. This clock is used as the reference clock for the PCIe* Tx/Rx circuitry and by the PCIe* core PLL to generate clocks for the PCIe* core logic.
REF_CLK14		I	LVTTL	BJ53	1			Reference Clock 14.31818 MHz Input. Single-ended 14.31818 MHz reference clock driven by a clock chip.
SATA_CLK100N SATA_CLK100P		I	LV Diff	AW53 AW55	2			Serial ATA 100 MHz Differential Clock. Reference clock 100 MHz Differential signal from a clock chip, provided separately from CLKIN_DMI, for use only as a 100 MHz source for SATA.

Table 32-24. Clock Input Interface Signals (Sheet 2 of 2)

Intel® Communications Chipset 89xx Series								
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)								
Signal Name		**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**							
UART_CLK		I	LVTTL	BN39	1		1149.1	UART Clock: This clock is passed to the baud clock generation logic for the UART. The clock can run at either 14.7456, or 48 MHz.
USB_CLK96N USB_CLK96P		I	LV Diff	AW5 AW3	2			USB* Clock 96 MHz. Differential reference input clock from an external clock chip.
TOTAL					15			

32.23 JTAG Interface

A hardware implementation is required if the customer wants to enable Boundary Scan for testing during manufacturing. Details on enabling JTAG are published in the PCH BSDL package. Contact your Intel Representative to request this package. This only applies to the JTAG chain in Table 32-25, "PCH JTAG Interface Signals".

Table 32-25. PCH JTAG Interface Signals

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/ External Resistor Pull-Up/ Down**	**BScan Support**	**Description**
JTCK	I	LVCMOS	BR29	1	External pull-down 51Ω 1% to VSS required		JTAG Test Clock for the JTAG controller.
JTDI	I	LVCMOS	BM30	1			JTAG Test Data In. Sampled with the rising edge of JTCK.
JTDO	OD	LVCMOS	BU29	1	External pull-up 51Ω 1% to VCCSUS required		JTAG Test Data Out.
JTMS	I	LVCMOS	BN31	1	External pull-up 51Ω 1% to VCCSUS required		JTAG Test Mode Select. Selects the state of the JTAG controller. Sampled with the rising edge of JTCK.
TOTAL				4			

Table 32-26. EndPoint JTAG Interface Signals (Sheet 1 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
EP_JTCK	I	LVCMOS	A12	1	External pull-down 51Ω 1% to VSS required		EndPoint JTAG Test Clock for the JTAG controller.
EP_JTDI	I	LVCMOS	Y15	1	Internal pull-up		EndPoint JTAG Test Data In. Sampled with the rising edge of EP_JTCK.
EP_JTDO	OD	LVCMOS	M15	1	External pull-up 51Ω 1% to VCCEPAUX required		EndPoint JTAG Test Data Out.

Table 32-26. EndPoint JTAG Interface Signals (Sheet 2 of 2)

Intel® Communications Chipset 89xx Series							
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)							
Signal Name **DH89xxCC & DH89xxCL**	**I/O Type**	**Technology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**BScan Support**	**Description**
EP_JTMS	I	LVCMOS	J17	1	External pull-up 51Ω 1% to VCCEPAUX required		EndPoint JTAG Test Mode Select. Selects the state of the JTAG controller. Sampled with the rising edge of EP_JTCK.
EP_JTRST#	I	LVCMOS	D13	1	Internal pull-up See Description		EndPoint JTAG Reset. Resets JTAG controller when asserted. The signal has an internal pull-up resistor to comply with 1149.1. An external 51Ω 1% pull-down resistor is required to disable JTAG and keep TAP in safe mode.
TOTAL				5			

32.24 Strapping Signals

Table 32-27. Strap Signals (Sheet 1 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GBE_AUX_PWR_AVAIL	RSVD137	I	LVTTL	B7	1	External pull-up required[1]	Rise Edge PCIE_EP_RST#	1149.1	**DH89xxCL:** • Reserved • No Connect **DH89xxCC:** Auxiliary Power Available. • This pin is a strapping option pin, latched at the rising edge of PCIE_EP_RST# or In-Band PCIe* Reset. • If this pin is driven high during init time, it indicates that auxiliary power is available and the device should support D3COLD power state if enabled to do so. • This pin value should be established before GBE_AUX_POWER_OK goes high.
GPIO8		I/O	LVTTL	BU16	1	Weak Internal pull-up for strap.	Rise Edge RSMRST#	1149.1	General Purpose I/O Port 8. Not Multiplexed. • This signal has a weak internal pull-up and must not be pulled low at boot up.
GPIO17		I/O	LVTTL	BF38	1	Platform Dependent	Rise Edge PWROK	1149.1	General Purpose I/O Port 17. Not Multiplexed. • This signal strapping sets the DMI termination voltage. • See PDG for additional information.

Table 32-27. Strap Signals (Sheet 2 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GPIO33		I/O	LVTTL	BN41	1	Weak Internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	General Purpose I/O Port 33. Not Multiplexed. **Flash Descriptor Security Overwrite.** • This signal is used to set the security override strap on the PCH. • If sampled low, the Flash Descriptor Security will be overridden. • If high, the security measures defined in the Flash Descriptor will be in effect. • This strap should only be enabled (pulled low) in manufacturing environments using an external pull-down resistor. **GPIO33:** 0 = Enable (Pull-Down Required) 1 = Disable (Default) ***Note:*** When the Security Overwrite is enable, it allows permission to every master to read and write to the entire Flash Components including areas outside the defined regions.
GPIO53		I/O	LVTTL	BM42	1	Platform Dependent. Weak internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	General Purpose I/O Port 53. Not Multiplexed. **DMI Coupling Strap**. • External platform dependent. **GPIO53:** 0 = AC Coupling (Pull-Down Required) 1 = DC Coupling (Default) See PDG for additional information.

Table 32-27. Strap Signals (Sheet 3 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GPIO55		I/O	LVTTL	BP45	1	Weak Internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	General Purpose I/O Port 55. Not Multiplexed. **Strap for BIOS Boot-Block Update Scheme.** • This mode allows the PCH to swap the Top-Block in the SPI (the boot block) with another location. **GPIO55:** 0 = Enable Top-Block Swap (Pull-down Required) 1 = Disable Top-Block Swap (Default) ***Note:*** The internal pull-up is disabled after PLTRST# de-asserts. If the signal is sampled low, this indicates that the system is strapped to the "Top-Block Swap" mode (the PCH inverts A16 for all cycles targeting BIOS space). • The status of this strap is readable via the Top Swap bit (Chipset Config Registers:Offset 3414h:bit 0). Software will not be able to clear the Top-Swap bit until the system is rebooted without GPIO55 being pulled down.
SIU0_DTR#		O	LVTTL	BP43	1	Weak Internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	UART Port 0 Data Terminal Ready: • When low, this pin informs the modem or data set that the UART port 0 is ready to establish a communication link. **SIW Configuration Port Address Select Strap:** • This strap selects the IO address for the SIW configuration port. Sampling occurs on the rising edge of PWROK. The straps are defined as follows: 0 = IO Addresses 02Eh and 02Fh (Pull-Down Required) 1 = IO Addresses 04Eh and 04Fh (default)

Table 32-27. Strap Signals (Sheet 4 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
BBS1 / GPIO51		I/O	LVTTL	BM38	1	Weak Internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	**BIOS Boot Strap 1.** BBS1 BBS0 0 0 NOT VALID 0 1 NOT VALID 1 0 NOT VALID 1 1 SPI BOOT If BBS1 interface is not used, the signals can be used as GPIO Port 51.
GBE0_LED	EP_SMB_ADR2	O	LVTTL	N6	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK See the ***Note*** in the DH89xxCC Description column	After assertion of AUX_PWROK (DH89xxCC) or EP_MAIN_PWR_OK (DH89xxCL)	1149.1	**DH89xxCL:** • This signal is used to Strap EP SMBus Slave Address bit 2 (High or Low). • The EP SMBus slave is accessed with address[7:1] = 1110_XX0 • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK **DH89xxCC:** • Port 0 LED. Programmable LED, mode encoding set by GbE EEPROM. • On power-up this signal becomes an input to Strap SMBus Slave Address bit 2 • The EP SMBus slave is accessed with address[7:1] = 1110_XX0. • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired.

Table 32-27. Strap Signals (Sheet 5 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GBE1_LED	EP_SMB_ADR3	O	LVTTL	R11	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK See the ***Note*** in the DH89xxCC Description column	After assertion of AUX_PWROK (DH89xxCC) or EP_MAIN_PWR_OK (DH89xxCL)	1149.1	**DH89xxCL:** • This signal is used to Strap EP SMBus Slave Address bit 3 (High or Low) • The EP SMBus slave is accessed with address[7:1] = 1110_XX0 • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK **DH89xxCC:** • Port 1 LED. Programmable LED, mode encoding set by GbE EEPROM. • On power-up this signal becomes an input to Strap SMBus Slave Address bit 3. • The EP SMBus slave is accessed with address[7:1] = 1110_XX0. • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired.

Table 32-27. Strap Signals (Sheet 6 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GBE2_LED	EP_VF_ENABLED	O	LVTTL	E4	1	Internal Pull-up Active for 200 µs after assertion of AUX_PWROK See the ***Note*** in the DH89xxCC Description column	After assertion of AUX_PWROK (DH89xxCC) or EP_MAIN_PWR_OK (DH89xxCL)	1149.1	**DH89xxCL:** • This signal can be use as a strapping function to enable or disable PCIe* SRIOV 0 = Disable SRIOV (External Pull-Down Required) 1 = Enable SRIOV (Default) • Sampling occurs during the first 200ns after assertion of EP_MAIN_PWR_OK **DH89xxCC:** • Port 2 LED. Programmable LED, mode encoding set by GbE EEPROM. • This signal can be use as a strapping function to enable or disable PCIe* SRIOV GBE2_LED: 0 = Disable SRIOV (External Pull-Down Required) 1 = Enable SRIOV (External Pull-up Required) • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as an LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired.

Table 32-27. Strap Signals (Sheet 7 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
GBE3_LED	EP_RESET_SEQ	O	LVTTL	U9	1	Internal Pull-up Active for 200us after assertion of AUX_PWROK See the ***Note*** in the DH89xxCC Description column	After assertion of AUX_PWROK (DH89xxCC) or EP_MAIN_PWR_OK (DH89xxCL)	1149.1	**DH89xxCL:** • This signal is a strap function that controls reset sequencing • This signal must be sampled as a 1 by allowing it to float or pulling it up 1 = PCIe* CEM 2.0 Compliant Reset sequence (Default) • Sampling occurs during the first 200 µs after assertion of EP_MAIN_PWR_OK **DH89xxCC:** • Port 3 LED. Programmable LED, mode encoding set by GbE EEPROM. • This signal can be use as a strapping function to control reset sequence. 0 = **Reserved** ***Note:*** • Should NOT be pulled LOW 1 = PCIe* CEM 2.0 Compliant Reset sequence (External Pull-up Required) • Sampling occurs during the first 200ns after assertion of AUX_PWROK. Thereafter the pin functions as a LED output. ***Note:*** • External board strap required to guarantee expected configuration setting because the internal pull-up is de-asserted before it is sampled. • For designs where GBEx_LED output signals are used to drive platform GBE LEDs, assert strap active until 1ms after assertion of AUX_PWROK then float. • For designs where GBEx_LED output signals are NOT used to drive platform GBE LEDs, the straps can be hardwired.

Table 32-27. Strap Signals (Sheet 8 of 8)

Intel® Communications Chipset 89xx Series									
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)									
Signal Name		**I/O Type**	**Tech nology**	**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/Down**	**Strap Qualifier**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**								
NRBOOTS		O	LVTTL	BM53	1	Weak Internal pull-down	Rise Edge PWROK when PLTRST# Asserted	1149.1	No Re-Boot Second Try Strap. See PDG for additional information.
BBS0		O	LVTTL	BJ41	1	Weak Internal pull-up	Rise Edge PWROK when PLTRST# Asserted	1149.1	BIOS Boot Strap 0. BBS1 BBS0 0 0 NOT VALID 0 1 NOT VALID 1 0 NOT VALID 1 1 SPI BOOT
TOTAL					14				

1. Refer to the platform design collateral for resistor values.

32.25 Reserved Signals

Note: All Reserved (RSVDx) signals are No Connects (NC), unless specified otherwise.

Table 32-28. Reserved Signals (Sheet 1 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)						
Signal Name		**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/ Pull-Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**					
RSVD0		E2	1		1149.1	Reserved Signal 0.
RSVD1		B5	1		1149.1	Reserved Signal 1.
RSVD2		B2	1		1149.1	Reserved Signal 2.
RSVD3		F1	1		1149.1	Reserved Signal 3.
RSVD4		D1	1		1149.1	Reserved Signal 4.
RSVD5		B15	1		1149.1	Reserved Signal 5.
RSVD6		C4	1		1149.1	Reserved Signal 6.
RSVD7		H17	1		1149.1	Reserved Signal 7.
RSVD8		A6	1		1149.1	Reserved Signal 8.
RSVD9		F18	1		1149.1	Reserved Signal 9.
RSVD10		M17	1		1149.1	Reserved Signal 10.
RSVD11		B17	1		1149.1	Reserved Signal 11.
RSVD12		P17	1		1149.1	Reserved Signal 12.
RSVD13		C16	1		1149.1	Reserved Signal 13.

Table 32-28. Reserved Signals (Sheet 2 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)						
Signal Name		Ball Map	Ball Count	Internal/External Resistor Pull-Up/ Pull-Down	BScan Support	Description
DH89xxCC	DH89xxCL					
RSVD14		J20	1		1149.1	Reserved Signal 14.
RSVD15		M20	1		1149.1	Reserved Signal 15.
RSVD16		F13	1		1149.1	Reserved Signal 16.
RSVD17		H22	1		1149.1	Reserved Signal 17.
RSVD18		F22	1		1149.1	Reserved Signal 18.
RSVD19		F23	1		1149.1	Reserved Signal 19.
RSVD20		R20	1		1149.1	Reserved Signal 20.
RSVD21		B21	1		1149.1	Reserved Signal 21.
RSVD22		D15	1		1149.1	Reserved Signal 22.
RSVD23		D23	1		1149.1	Reserved Signal 23.
RSVD24		L20	1		1149.1	Reserved Signal 24.
RSVD25		B23	1		1149.1	Reserved Signal 25.
RSVD26		E17	1		1149.1	Reserved Signal 26.
RSVD27		A22	1		1149.1	Reserved Signal 27.
RSVD28		C19	1		1149.1	Reserved Signal 28.
RSVD29		B25	1		1149.1	Reserved Signal 29.
RSVD30		F20	1		1149.1	Reserved Signal 30.
RSVD31		D25	1		1149.1	Reserved Signal 31.
RSVD32		H27	1		1149.1	Reserved Signal 32.
RSVD33		M22	1		1149.1	Reserved Signal 33.
RSVD34		E21	1		1149.1	Reserved Signal 34.
RSVD35		H25	1		1149.1	Reserved Signal 35.
RSVD36		R15	1			Reserved Signal 36
RSVD37		U15	1			Reserved Signal 37
RSVD38		BT31	1	External pull-up required[1]		Reserved Signal 38. • This signal needs a pull-up to VCCSUS.
RSVD39		BG25	1	External pull-down required 1KΩ to VSS		Reserved Signal 39.
RSVD41		AE46	1			Reserved Signal 41.
RSVD42		AB46	1			Reserved Signal 42.
RSVD43		BA46	1			Reserved Signal 43.
RSVD44		BA44	1			Reserved Signal 44.
RSVD45		AT3	1			Reserved Signal 45.
RSVD46		AR2	1			Reserved Signal 46.

Table 32-28. Reserved Signals (Sheet 3 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)						
Signal Name		Ball Map	Ball Count	Internal/External Resistor Pull-Up/ Pull-Down	BScan Support	Description
DH89xxCC	DH89xxCL					
RSVD47		BC49	1		1149.1	Reserved Signal 47.
RSVD48		BE56	1		1149.1	Reserved Signal 48.
RSVD49		BE54	1		1149.1	Reserved Signal 49.
RSVD50		BF55	1		1149.1	Reserved Signal 50.
RSVD51		D55	1			Reserved Signal 51.
RSVD52		C54	1			Reserved Signal 52.
RSVD53		BT25	1	Weak Internal pull-down	1149.1	Reserved Signal 53.
RSVD54		E56	1			Reserved Signal 54.
RSVD55		D57	1			Reserved Signal 55.
RSVD56		BU6	1			Reserved Signal 56.
RSVD57		BT5	1			Reserved Signal 57.
RSVD59		BU4	1			Reserved Signal 59.
RSVD60		BP5	1			Reserved Signal 60.
RSVD61		M41	1			Reserved Signal 61.
RSVD62		P43	1			Reserved Signal 62.
RSVD63		AE8	1			Reserved Signal 63.
RSVD64		AB8	1			Reserved Signal 64.
RSVD65		BP1	1			Reserved Signal 65.
RSVD66		BM1	1			Reserved Signal 66.
RSVD67		BP3	1			Reserved Signal 67.
RSVD68		BN2	1			Reserved Signal 68.
RSVD69		L25	1			Reserved Signal 69.
RSVD70		M25	1			Reserved Signal 70.
RSVD71		P25	1	External 100pF cap to ground.		Reserved Signal 71. • Connect to 100pF capacitor tied to ground.
RSVD72		L27	1	External Pull-down required <1KΩ to VSS		Reserved Signal 72. • This pin must be tied to ground or pull to ground through a resistor.
RSVD73		BM5	1		1149.1	Reserved Signal 73.
RSVD74		BH8	1		1149.1	Reserved Signal 74.
RSVD75		BJ46	1		1149.1	Reserved Signal 75.
RSVD76		BK46	1		1149.1	Reserved Signal 76.
RSVD77		BT33	1			Reserved Signal 77.
RSVD78		BJ15	1		1149.1	Reserved Signal 78.

Table 32-28. Reserved Signals (Sheet 4 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)						
Signal Name		**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/ Pull-Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**					
RSVD79		C9	1			Reserved Signal 79
RSVD80		E11	1			Reserved Signal 80
RSVD81		B11	1			Reserved Signal 81
RSVD82		E9	1			Reserved Signal 82
RSVD83		J1	1			Reserved Signal 83
RSVD84		H6	1			Reserved Signal 84
RSVD85		F5	1			Reserved Signal 85
RSVD86		H8	1			Reserved Signal 86
RSVD87		Y14	1			Reserved Signal 87
RSVD88		R14	1			Reserved Signal 88
RSVD89		J15	1			Reserved Signal 89
RSVD90		L15	1			Reserved Signal 90
Refer to Table 32-4 for DH89xxCC Signals	RSVD91	W3	1			DH89xxCL Reserved Signals: ***Note:*** • Unless termination is specified for a specific signal, all Reserved signals are No Connects (NC)
	RSVD92	AA2	1			
	RSVD93	AC2	1			
	RSVD94	AB11	1			
	RSVD95	AA5	1			
	RSVD96	AB3	1			
	RSVD97	AC4	1			
	RSVD98	AB12	1			
	RSVD99	AG5	1			
	RSVD100	AJ3	1			
	RSVD101	AL2	1			
	RSVD102	AN2	1			
	RSVD103	AJ5	1			
	RSVD104	AL5	1			
	RSVD105	AM3	1			
	RSVD106	AN4	1			
Refer to Table 32-24 for DH89xxCC Signals	RSVD107	AF1	1			
	RSVD108	AF3	1			
Refer to Table 32-4 for DH89xxCC Signals	RSVD109	AG8	1			
	RSVD110	AE11	1			

Table 32-28. Reserved Signals (Sheet 5 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920) DH89xxCL: (8925 ≤ *SKU* ≤ 8955)						
Signal Name		**Ball Map**	**Ball Count**	**Internal/External Resistor Pull-Up/ Pull-Down**	**BScan Support**	**Description**
DH89xxCC	**DH89xxCL**					
Refer to Table 32-7 for DH89xxCC Signals	RSVD113	U11	1	External pull-up resistor to VCCPEP3P3AUX is required		DH89xxCL Reserved Signals: ***Note:*** • Unless termination is specified for a specific signal, all Reserved signals are No Connects (NC)
	RSVD114	Y12	1	External pull-up resistor to VCCPEP3P3AUX is required		
	RSVD115	U12	1	External pull-up resistor to VCCPEP3P3AUX is required		
Refer to Table 32-5 for DH89xxCC Signals	RSVD116	T3	1			
	RSVD117	U2	1			
	RSVD118	N2	1			
	RSVD119	Y8	1			
	RSVD120	R8	1			
	RSVD121	U8	1			
	RSVD122	L2	1			
	RSVD123	N4	1			
Refer to Table 32-8 for DH89xxCC Signals	RSVD124	Y9	1			
	RSVD125	M8	1			
	RSVD126	R2	1			
	RSVD127	J3	1			
	RSVD128	R6	1			
	RSVD129	G4	1			
	RSVD130	R4	1			
	RSVD131	H10	1			
Refer to Table 32-4for DH89xxCC Signals	RSVD132	M3	1			
	RSVD133	C6	1			
	RSVD134	F3	1			
	RSVD135	M9	1			
Note: RSVD40, RSVD58, RSVD111, and RSVD112 are not listed in the RSVD sequence numbering. They do not exist in either DH89xxCC or DH89xxCL.						
TOTAL			132			

1. Refer to the platform design collateral for resistor values.

32.26 Power and Ground Signals

Note: Refer to Section 35.2.1, "Ball Map Pin Lists" for Ground (VSS) Ball Map.

Table 32-29. Power and Ground Signals (Sheet 1 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920)		DH89xxCL: (8925 ≤ *SKU* ≤ 8955)		Ball Map	Ball Count	Description
Voltage Rail	Signal Name	Voltage Rail	Signal Name			
1.00 V	VCC	1.05 V	VCC	AN24 AN28 AN32 AR26 AR30 AR32 AR34 AR36 AU26 AU30 AU34	11	Power supply for core logic. Will be off in S3/S4/S5
1.00 V	VCCA_SATA	1.05 V	VCCA_SATA	AV41 AV44 BA43 BC43	4	Analog Power supply for SATA I/O interface.
1.00 V	VCCA_USB	1.05 V	VCCA_USB	BC11 BC12	2	Analog Power supply for USB* level shifter.
1.00 V	VCCAEP_PE	1.05 V	VCCAEP_PE	Y38 Y41 AA38 AB41 AB43 AC38 AE36 AE40 AE43 AG38 AG41 AG43	12	Analog Power supply EndPoint for PCI Express I/O interface. Analog power.April 2014

Table 32-29. Power and Ground Signals (Sheet 2 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920)		DH89xxCL: (8925 ≤ *SKU* ≤ 8955)		Ball Map	Ball Count	Description
Voltage Rail	Signal Name	Voltage Rail	Signal Name			
1.00 V	VCCEP	1.05 V	VCCEP	U31 U36 V27 V33 Y28 Y30 Y34 AA28 AA30 AA34 AC30 AC34 AE28 AE32 AG26 AG30 AG34 AJ28 AJ32 AJ36 AL26 AL30 AL34	23	Power supply EndPoint Unit for core logic. Will be off in S3/S4/S5
1.00 V	VCCEP1P0_CRU	1.05 V	VCCEP1P0_CRU	AC24 AC26	2	Digital Power supply EndPoint Unit for CRU circuitry (CRU PLL and REFCLK inputs)
1.00 V	VCCEPAUX	1.05 V	VCCEPAUX	U25 V22 Y22 AA22 AB17 AC20 AE22 AE24 AG20 AG24 AJ22 AJ24 AL20	13	Auxiliary Power supply EndPoint Unit for core logic
varies[1]	VCCPCPU	varies[1]	VCCPCPU	BC41	1	Power supply for CPU I/O Signals: This voltage can be 1.0V or 1.05V based on the processor used. Powered by the same supply as the processor I/O voltage. This supply is used to drive the processor interface signals. • See the processor documentation for the appropriate voltage level.

Table 32-29. Power and Ground Signals (Sheet 3 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920)		DH89xxCL: (8925 ≤ *SKU* ≤ 8955)		Ball Map	Ball Count	Description
Voltage Rail	Signal Name	Voltage Rail	Signal Name			
1.00 V	VCCSUS	1.05 V	VCCSUS	AR22 AU22 AV20 AV24 BA22 BC20 BD20 BD22	8	Sustain power supply for core logic. Will be off in G3 only
1.00 V	VCCSUS_USB	1.05 V	VCCSUS_USB	AV8 BA8	2	Sustain power supply for USB*2 High Speed TX/RX interface and control Logic.
1.05 V	VCC1P05	1.05 V	VCC1P05	AN8 AN9	2	Power supply for internal reference. This voltage can not be changed from the recommended value.
varies[1]	VCCA_DMI	varies[1]	VCCA_DMI	AL38 AL41 AL44 AN38 AN41 AN44 AR38 AT40 AT43	9	Analog Power supply for DMI. This voltage can be 1.0V or 1.05V based on the processor used. Powered by the same supply as the processor I/O voltage. See the respective processor documentation to find the appropriate voltage level. This power will be shut down in S3, S4, and S5 states.
1.05 V	**VCCAEPAUX[2]**	1.05 V	**VCCAEP_1P05[2]**	U25 V22 Y22 AA22 AB17 AC20 AE22 AE24 AG20 AG24 AJ22 AJ24 AL20	13	**DH89xxCL:** • Analog Power supply for EndPoint Unit. Supply voltage of EndPoint Unit I/O interface. **DH89xxCC:** • Analog Auxiliary Power supply EndPoint Unit. Supply voltage for Gigabit Ethernet Core, and I/O interface EndPoint Unit.
1.50 V	VCCA1P5_TS0	1.50 V	VCCA1P5_TS0	AN17	1	Analog Power supply for Thermal Sensor 0.
1.50 V	VCCA1P5_TS1	1.50 V	VCCA1P5_TS1	AN15	1	Analog Power supply for Thermal Sensor 1.
1.80 V	VCC1P8	1.80 V	VCC1P8	BC27 BC33 BC36 BD31	4	Power supply for core well I/O buffers. This power will be off in S3/S4/S5 or G3
1.80 V	VCCA1P8_DMI	1.80 V	VCCA1P8_DMI	AG44	1	Analog Power supply for DMI PLL and PHY Compensation Logic. • This power is supplied with core well

Table 32-29. Power and Ground Signals (Sheet 4 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920)		DH89xxCL: (8925 ≤ *SKU* ≤ 8955)		Ball Map	Ball Count	Description
Voltage Rail	Signal Name	Voltage Rail	Signal Name			
1.80 V	VCCA1P8_SATA	1.80 V	VCCA1P8_SATA	AV46	1	Analog Power supply for the SATA PLL and PHY Compensation Logic.
1.80 V	VCCA1P8_TS0	1.80 V	VCCA1P8_TS0	AN34	1	Analog Power supply for Thermal Sensor 0.
1.80 V	VCCA1P8_TS1	1.80 V	VCCA1P8_TS1	AN11	1	Analog Power supply for Thermal Sensor 1.
1.80 V	VCCA1P8_USB	1.80 V	VCCA1P8_USB	BA9 BC8	2	Analog Power supply for USB* PLL and PHY Compensation Logic.
1.80 V	VCCAEP1P8_CRU	1.80 V	VCCAEP1P8_CRU	R22 U22	2	Analog Power supply EndPoint Unit for CRU circuitry (CRU PLL and REFCLK inputs).
1.80 V	VCCAEP1P8_PE	1.80 V	VCCAEP1P8_PE	Y43	1	Analog Power supply EndPoint for PCI Express PLL and PHY Compensation Logic.
1.80 V	VCCAEP1P8AUX	1.80 V	VCCAEP1P8AUX	AG9 AG11 AL8	3	Analog Auxiliary Power supply EndPoint Unit for I/O Logic, Gigabit Ethernet PLL and PHY Compensation Logic. This power is not expected to be shut off unless the system is unplugged
1.80 V	VCCASUS1P8_USB	1.80 V	VCCASUS1P8_USB	AT11	1	Analog Sustain Power supply for USB* band gap
1.80 V	VCCSUS1P8	1.80 V	VCCSUS1P8	AV14 AV17 BA17	3	Sustain power supply for I/O logic. This power is not expected to be shut off unless the system is in G3 or unplugged
1.80 V	VCCSUS1P8_USB	1.80 V	VCCSUS1P8_USB	AN12 AT14 AT15 AT17	4	Sustain Power supply for USB* TX/RX interface and control Logic
3.30 V	VCC3P3_RTC	3.30 V	VCC3P3_RTC	BF27	1	3.3V RTC power supply. This power can drop to 2.0V(min) in G3 state when power is supplied by the coin cell battery. ***Note:*** • Implementations should not attempt to clear CMOS by using a jumper to pull VCC3P3_RTC low. Clearing CMOS in a PCH-platform can be done by using a jumper on RTCRST#.

Table 32-29. Power and Ground Signals (Sheet 5 of 5)

Intel® Communications Chipset 89xx Series						
DH89xxCC: (8900 ≤ *SKU* ≤ 8920)		DH89xxCL: (8925 ≤ *SKU* ≤ 8955)		Ball Map	Ball Count	Description
Voltage Rail	Signal Name	Voltage Rail	Signal Name			
3.30 V	VCC3P3	3.30 V	VCC3P3	AV26 AV30 AV34 AV36 AV38 AY27 AY33 BA25 BA31 BA36 BA38 BC25	12	Power supply for core well I/O buffers. This power will be off in S3/S4/S5 or G3
3.30 V	VCCPEP3P3AUX	3.30 V	VCCPEP3P3AUX	AB15 AE14 AE17 AG17	4	Auxiliary Power supply EndPoint Unit for I/O Logic. This power is not expected to be shut off unless the system is unplugged
3.30 V	VCCSUS3P3	3.30 V	VCCSUS3P3	BA15 BC14 BC15 BF17	4	Sustain power supply for I/O logic. This power is not expected to be shut off unless the system is in G3 or unplugged
3.30 V	VCCSUS3P3_RTC	3.30 V	VCCSUS3P3_RTC	BF25	1	Sustain power supply for RTC Block: Suspend well power for part of the RTC logic. ***Note:*** Can be connected to VCCSUS3P3 on the board
3.30 V	VCCSUS3P3_USB	3.30 V	VCCSUS3P3_USB	AT8 AV9 AV11 AV12 BA12	5	Sustain Power supply for USB* classic TX output interface
GND	VSS	GND	VSS	Refer to Section 35.2.1, "Ball Map Pin Lists" for VSS Ball Map	358	Ground
GND	VSSA_USB	GND	VSSA_USB	AT9	1	Quiet Analog Ground for USB*
TOTAL					514	

1. The voltage can be 1.0V or 1.05V based on the processor used.
2. Different Power rail naming between DH89xxCC and DH89xxCL.

§ §

33.0 Electrical Characteristics

This chapter contains DC and AC characteristics. AC timing diagrams are included.

Note: Icc values in this section are pre-silicon estimates.

33.1 Absolute Maximum and Minimum Ratings

Table 33-1 specifies the absolute maximum and minimum ratings of the PCH. At conditions outside of the functional operating condition limits, but within the absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. If a device is returned to conditions within the functional operating limits after having been subjected to conditions outside these limits (but within the absolute maximum and minimum ratings) the device may be functional, but with its lifetime degraded depending on exposure to conditions exceeding the functional operating condition limits.

At conditions exceeding the absolute maximum and minimum ratings, neither functionality nor long-term reliability can be expected. Moreover, if a device is subjected to these conditions for any length of time, it will either not function or its reliability will be severely degraded when returned to conditions within the functional operating condition limits.

Although the PCH contains protective circuitry to resist damage from Electro-Static Discharge (ESD), precautions should always be taken to avoid high static voltages or electric fields

Table 33-1. PCH Absolute Maximum and Minimum Ratings

Symbol	Parameter	Minimum Limits	Maximum Limits	Units
$T_{j\ max}$	Junction Temperature	0	103	°C
$T_{absolute\ storage}$	Device storage when exceeded for any length of time	-25	125	°C
Tshort term storage	The ambient storage temperature and time for up to 72 hours.	-25	85	°C
Tsustained storage Time and Temp.	The ambient storage temperature and time for up to 30 months.	-5	40	°C
RH sustained storage	The maximum device storage relative humidity for up to 30 months.		60% at 24°C	

33.2 Recommended Power Supply Range

Note: In the following table, Rail specifications for the DH89xxCC and DH89xxCL are differentiated with the DH89xxCC specification above the DH89xxCL specification. Otherwise, the specification is the same for all SKUs.

Table 33-2. Operating Conditions Power Supply Rail (Sheet 1 of 2)

Symbol	Tolerance	Min	Nominal	Max	Unit
VCC[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCA_SATA[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCA_USB[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCAEP_PE[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCEP[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCEP1P0_CRU[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCEPAUX[1]	+/-5%	0.95	1.00[1]	1.05	V
		1.00	1.05[1]	1.10	
VCCPCPU[1,2,3]	+/-5%	0.91	0.95	1.05[2]	V
				1.10[2]	
VCCSUS[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCSUS_USB[1]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCCA_DMI[1,3]	+/-5%	0.95	1.00	1.05	V
		1.00	1.05	1.10	
VCC1P05	+/-5%	1.00	1.05	1.10	V
VCCAEPAUX[5]	+/-5%	1.00	1.05	1.10	V
VCCAEP_1P05[5]					
VCCA1P5_TS0	+/-5%	1.42	1.50	1.57	V
VCCA1P5_TS1	+/-5%	1.42	1.50	1.57	V
VCC1P8	+/-5%	1.71	1.80	1.89	V
VCCA1P8_DMI	+/-5%	1.71	1.80	1.89	V
VCCA1P8_SATA	+/-5%	1.71	1.80	1.89	V
VCCA1P8_TS0	+/-5%	1.71	1.80	1.89	V
VCCA1P8_TS1	+/-5%	1.71	1.80	1.89	V
VCCA1P8_USB	+/-5%	1.71	1.80	1.89	V
VCCAEP1P8_CRU	+/-5%	1.71	1.80	1.89	V
VCCAEP1P8_PE	+/-5%	1.71	1.80	1.89	V
VCCAEP1P8AUX	+/-5%	1.71	1.80	1.89	V

Table 33-2. Operating Conditions Power Supply Rail (Sheet 2 of 2)

Symbol	Tolerance	Min	Nominal	Max	Unit
VCCASUS1P8_USB	+/-5%	1.71	1.80	1.89	V
VCCSUS1P8	+/-5%	1.71	1.80	1.89	V
VCCSUS1P8_USB	+/-5%	1.71	1.80	1.89	V
VCC3P3	+/-5%	3.13	3.30	3.46	V
VCCPEP3P3AUX	+/-5%	3.13	3.30	3.46	V
VCCSUS3P3	+/-5%	3.13	3.30	3.46	V
VCCSUS3P3_RTC	+/-5%	3.13	3.30	3.46	V
VCCSUS3P3_USB	+/-5%	3.13	3.30	3.46	V
VCC3P3_RTC[4]	+/-5%	2.00	-	3.46	V
VCC3P3_RTC (Battery)		2.00	-	3.46	V
VSSA_USB	+/-5%		GND		V

1. DH89xxCC operating voltage is 1.00V.
   DH89xxCL operating voltage is 1.05V.
2. DH89xxCC maximum voltage is 1.05V.
   DH89xxCL maximum voltage is 1.10V.
3. Analog power supply VCCA_DMI & VCCPCPU. This voltage is based on the type of processor. See the processor documentation for the appropriate voltage level. This power is shut down in S3, S4, and S5 states.
4. VCC3P3_RTC is the voltage applied to the VCC3P3_RTC well of the PCH. When the system is in G3 state (Mechanical Off), this is supplied by the coin cell battery (VCC3P3_RTC(Battery)). For sleep states (Sx: S0 thru S5), this is supplied by VCCSUS3P3.
5. VCCAEPAUX & VCCAEP_1P05 are applicable only to the DH89xxCC & DH89xxCL, respectively.

33.3 Maximum ICC Supply Current

Table 33-3. Maximum Supply Current (Sheet 1 of 2)

DH89xxCC			DH89xxCL		
Well/Voltage Rail	Symbol	ICC (max)	Well/Voltage Rail	Symbol	ICC (max)
1.00 V	VCC	2000 mA	1.05 V	VCC	2000 mA
1.00 V	VCCA_SATA	300 mA	1.05 V	VCCA_SATA	300 mA
1.00 V	VCCA_USB	120 mA	1.05 V	VCCA_USB	120 mA
1.00 V	VCCAEP_PE	1600 mA	1.05 V	VCCAEP_PE	1600 mA
1.00 V	VCCEP	10.0 A	1.05 V	VCCEP	21.0 A
1.00 V	VCCEP1P0_CRU	100 mA	1.05 V	VCCEP1P0_CRU	100 mA
1.00 V	VCCEPAUX	1500 mA	1.05 V	VCCEPAUX	100 mA
varies[1]	VCCPCPU	50 mA	varies[1]	VCCPCPU	50 mA
1.00 V	VCCSUS	520 mA	1.05 V	VCCSUS	520 mA
1.00 V	VCCSUS_USB	50 mA	1.05 V	VCCSUS_USB	50 mA
1.05 V	VCC1P05	35 mA	1.05 V	VCC1P05	35 mA
varies[1]	VCCA_DMI	800 mA	varies[1]	VCCA_DMI	800 mA
1.05 V	VCCAEPAUX	550 mA	1.05 V	VCCAEP_1P05	550 mA
1.50 V	VCCA1P5_TS0	10 mA	1.50 V	VCCA1P5_TS0	10 mA
1.50 V	VCCA1P5_TS1	10 mA	1.50 V	VCCA1P5_TS1	10 mA
1.80 V	VCC1P8	50 mA	1.80 V	VCC1P8	50 mA

Table 33-3. Maximum Supply Current (Sheet 2 of 2)

DH89xxCC			DH89xxCL		
Well/Voltage Rail	Symbol	ICC (max)	Well/Voltage Rail	Symbol	ICC (max)
1.80 V	VCCA1P8_DMI	55 mA	1.80 V	VCCA1P8_DMI	55 mA
1.80 V	VCCA1P8_SATA	55 mA	1.80 V	VCCA1P8_SATA	55 mA
1.80 V	VCCA1P8_TS0	30 mA	1.80 V	VCCA1P8_TS0	30 mA
1.80 V	VCCA1P8_TS1	30 mA	1.80 V	VCCA1P8_TS1	30 mA
1.80 V	VCCA1P8_USB	45 mA	1.80 V	VCCA1P8_USB	45 mA
1.80 V	VCCAEP1P8_CRU	90 mA	1.80 V	VCCEAP1P8_CRU	90 mA
1.80 V	VCCAEP1P8_PE	55 mA	1.80 V	VCCAEP1P8_PE	55 mA
1.80 V	VCCAEP1P8AUX	100 mA	1.80 V	VCCAEP1P8AUX	100 mA
1.80 V	VCCASUS1P8_USB	15 mA	1.80 V	VCCASUS1P8_USB	15 mA
1.80 V	VCCSUS1P8	50 mA	1.80 V	VCCSUS1P8	50 mA
1.80 V	VCCSUS1P8_USB	250 mA	1.80 V	VCCSUS1P8_USB	250 mA
3.00 V	VCC3P3_RTC (Battery)	6 µA	3.00 V	VCC3P3_RTC (Battery)	6 µA
3.30 V	VCC3P3	150 mA	3.30 V	VCC3P3	150 mA
3.30 V	VCC3P3_RTC	3 mA	3.30 V	VCC3P3_RTC	3 mA
3.30 V	VCCPEP3P3AUX	150 mA	3.30 V	VCCPEP3P3AUX	150 mA
3.30 V	VCCSUS3P3	150 mA	3.30 V	VCCSUS3P3	150 mA
3.30 V	VCCSUS3P3_RTC	3 mA	3.30 V	VCCSUS3P3_RTC	3 mA
3.30 V	VCCSUS3P3_USB	90 mA	3.30 V	VCCSUS3P3_USB	90 mA

1. The voltage can be 1.0v or 1.05v based on the processor used.

33.4 Power Supply Pin Groupings

The following tables are suggestions of how to group power pins. These groupings help to minimize the number of power regulators required to implement power requirements. Table 33-4, "Power for Standby Support Designs" is for designs that require standby functionality for systems that go into power saving modes. Table 33-5, "Power for Non-Standby Support Designs" is for system designs that are meant to be either ON or OFF and do not go into power saving modes.

Table 33-4. Power for Standby Support Designs (Sheet 1 of 2)

Customer Name	Board Regulator Grouping[1]	Domain	Normal Mode
VCCAEP1P8_PE	1	Core	EndPoint
VCCAEP1P8_CRU	1	Core	EndPoint
VCCA1P8_DMI	1	Core	Non-EndPoint
VCCA1P8_SATA	1	Core	Non-EndPoint
VCCA1P8_USB	1	Core	Non-EndPoint
VCCA1P8_TS0	1	Core	Non-EndPoint
VCCA1P8_TS1	1	Core	Non-EndPoint

Table 33-4. Power for Standby Support Designs (Sheet 2 of 2)

Customer Name	Board Regulator Grouping[1]	Domain	Normal Mode
VCC1P8	1	Core	Non-EndPoint
VCCAEP_PE	2	Core	EndPoint
VCCEP1P0_CRU	2	Core	EndPoint
VCCA_SATA	2	Core	Non-EndPoint
VCCA_USB	2	Core	Non-EndPoint
VCC	2	Core	Non-EndPoint
VCCEP	2	Core	EndPoint
VCCAEP1P8AUX	3	Aux	EndPoint
VCCASUS1P8_USB	3	Sus	Non-EndPoint
VCCSUS1P8_USB	3	Sus	Non-EndPoint
VCCSUS1P8	3	Sus	Non-EndPoint
VCCAEPAUX	4	Aux	EndPoint
VCCPEP3P3AUX	5	Aux	EndPoint
VCCSUS3P3_USB	5	Sus	Non-EndPoint
VCCSUS3P3_RTC	5	Sus	Non-EndPoint
VCCSUS3P3	5	Sus	Non-EndPoint
VCCA_DMI	6	Core	Non-EndPoint[2]
VCC1P05	7	Core	Non-EndPoint.
VCCPCPU	8	Core	Non-EndPoint. Connect to CPU IO rail
VCCEPAUX[3]	9	Aux	EndPoint
VCCAEP_1P05[3]	9	Core	EndPoint
VCCSUS_USB	9	Sus	Non-EndPoint
VCCSUS	9	Sus	Non-EndPoint
VCCA1P5_TS0	10	Core	Non-EndPoint
VCCA1P5_TS1	10	Core	Non-EndPoint
VCC3P3	11	Core	Non-EndPoint
VCC3P3_RTC	Battery	RTC	Non-EndPoint
VSS	G	GND	EndPoint and Non-EndPoint

1. Board Regulator Grouping refers to the minimum number of regulators that can be used for adequate operation.
2. Voltage level depends on processor use in your design.
3. VCCAEPAUX & VCCAEP_1P05 are applicable only to the DH89xxCC & DH89xxCL, respectively.

Table 33-5. Power for Non-Standby Support Designs (Sheet 1 of 2)[1]

Pin Name	Board Regulator Grouping[2]	Domain	Normal Mode
VCCAEP1P8_PE	1	Main	EndPoint
VCCAEP1P8_CRU	1	Main	EndPoint
VCCA1P8_DMI	1	Main	Non-EndPoint
VCCA1P8_SATA	1	Main	Non-EndPoint
VCCA1P8_USB	1	Main	Non-EndPoint
VCCA1P8_TS0	1	Main	Non-EndPoint

Table 33-5. Power for Non-Standby Support Designs (Sheet 2 of 2)[1]

Pin Name	Board Regulator Grouping[2]	Domain	Normal Mode
VCCA1P8_TS1	1	Main	Non-EndPoint
VCC1P8	1	Main	Non-EndPoint
VCCAEP1P8AUX	1	Main	EndPoint
VCCASUS1P8_USB	1	Main	Non-EndPoint
VCCSUS1P8_USB	1	Main	Non-EndPoint
VCCSUS1P8	1	Main	Non-EndPoint
VCCAEP_PE	2	Main	EndPoint
VCCEP1P0_CRU	2	Main	EndPoint
VCCA_SATA	2	Main	Non-EndPoint
VCCA_USB	2	Main	Non-EndPoint
VCC	2	Main	Non-EndPoint
VCCEPAUX	2	Main	EndPoint
VCCSUS_USB	2	Main	Non-EndPoint
VCCSUS	2	Main	Non-EndPoint
VCCEP	2	Main	EndPoint
VCCAEPAUX[4]	3	Main	EndPoint
VCCAEP_1P05[4]	3	Main	EndPoint
VCCA_DMI	4	Main	Non-EndPoint [3]
VCC1P05	5	Main	Non-EndPoint
VCCPCPU	6	Main	Non-EndPoint. Connect to CPU IO rail
VCCPEP3P3AUX	7	Main	EndPoint
VCCSUS3P3_USB	7	Main	Non-EndPoint
VCCSUS3P3_RTC	7	Main	Non-EndPoint
VCCSUS3P3	7	Main	Non-EndPoint
VCC3P3	7	Main	Non-EndPoint
VCC3P3_RTC	7	Main	Non-EndPoint
VCCA1P5_TS0	8	Main	Non-EndPoint
VCCA1P5_TS1	8	Main	Non-EndPoint
VSS	G	GND	EndPoint and Non-EndPoint

1. For non-standby support, you can connect Suspend [SUS] and Auxiliary [AUX] supplies together.
2. Board Regulator Grouping refers to the minimum number of regulators that can be used for adequate operation.
3. Voltage level depends on processor use in your design.
4. VCCAEPAUX & VCCAEP_1P05 are applicable only to the DH89xxCC & DH89xxCL, respectively.

33.5 General DC Characteristics

Table 33-6. DC Input Characteristic Signal Association

Symbol	Associated Signals
VIH1/VIL1	**SMBus Signals:** EP_SMBCLK, EP_SMBDAT **SMBus GbE Signals:** GBE_SMBCLK, GBE_SMBDAT, GBE_SMBALRT# **GbE Signals:** SFP[3:0]_I2C_DATA, SRDS[3:0]_SD, GBE[3:0]_LED, GBE[3:0]_SWDP[1:0], GBE_AUX_PWR_OK, GBE_EE_DO **PCIe* Signals:** PCIE_EP_RST **Power Management Signals:** EP_CRU_EN, EP_MAIN_PWR_OK, GBE_AUX_PWR_AVAIL, GBE_AUX_PWR_OK
VIH2/VIL2	**Clock Signals:** REF_CLK14, PCICLK, UART_CLK **Processor Signals:** A20GATE, RCIN# **Interrupt Signals:** SERIRQ **Power Management Signals:** BMBUSY# **UART Signals:** SIU[1:0]_RXD, SIU[1:0]_DSR#, SIU[1:0]_DCD#, SIU[1:0]_CTS#, SIU[1:0]_RI# **SATA Signals:** SATA4_GP, SATA5_GP **SPI Signals:** SPI_MISO **LPC/Firmware Hub Signals:** LAD[3:0], LDRQ0#, LDRQ1# **GPIO Signals:** GPIO[55, 54, 53, 52, 51, 50, 49, 48, 39, 38, 37, 36, 35, 34, 33, 32, 23, 22, 21, 20, 19, 18, 17,16, 7, 6, 5, 4, 3, 2, 1, 0] **Strap Signals:** SPI_MOSI, BBS[1:0]#
VIH3/VIL3	**SMBus Signals:** MST_SMBCLK, MST_SMBDAT **System Management Link Signals:** SML1CLK, SML1DAT **Power Management Signals:** PWRBTN#, RI#, SYS_RESET#, WAKE#, SYS_PWROK, MEPWROK **GPIO Signals:** GPIO[75, 74, 73, 72, 63, 62, 61, 60, 59, 58, 57, 56, 47, 46, 45, 44, 43, 42, 41, 40, 31, 30, 28, 27, 26, 25, 24, 15, 14, 13, 12, 11, 10, 9, 8] **USB Signals:** OC[3:0]#
VIH4/VIL4	**JTAG Signals:** EP_JTDI, EP_JTMS, EP_JTRST#, EP_JTCK
VIH5/VIL5	**JTAG Signals:** JTDI, JTMS, JTCK
VIH6/VIL6	**Processor Signals:** THRMTRIP#
VIMIN7/VIMAX7	**PCI Express* Data RX Signals:** PCIE_EP_RX[p,n][15:0], PCIE_RC_RX[p,n][3:0], DMI_RX[p,n][3:0]
VIH8/VIL8	**Real Time Clock Signals**: RTCX1 (Input from Clock Generator)
VIMIN9 - Gen1i, m VIMAX9 - Gen1i. m VIMIN9 - Gen2i, m VIMAX9 - Gen2i. m	**SATA RX Signals:** SATA[5:4]_RX[P,N]
Rin-Diff10	**GbE Signals:** SRDSI_[3:0]_[P,N]
VIH11 (Absolute Maximum) VIL11 (Absolute Minimum) Vclk_in_cross (abs)	**Clock Signals**: CRU_CLK100[P,N], DMI_CLK100[P,N], USB_CLK96[P,N], SATA_CLK100[P,N], PCIE_EP_CLK100[P,N], GBE_CLK100[P,N]
VIH12/VIL12	**Power Management Signals:** PWROK, RSMRST# **System Management Signals:** INTRUDER# **Reset Signals:** SRTCRST#
VIH13/VIL13	**Reset Signals:** RTCRST#
VIH_PECI/VIL_PECI	**Thermal Reporting Signals**: PECI
VDI / VCM / VSE	**USB Signals:** USB[5:0]D[p,n] (Low-speed and Full-speed)[1]
V_{HSSQ} / V_{HSDSC} / V_{HSCM}	**USB Signals:** USB[5:0]D[p,n] (in High-speed Mode)[1]

1. USB not 5V tolerant

Table 33-7. DC Input Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
VIL1	Input Low Voltage	–0.3	0.8	V	
VIH1	Input High Voltage	2.0	VCCPEP3P3AUX+0.3	V	
VIL2	Input Low Voltage	–0.3	0.8	V	
VIH2	Input High Voltage	2.0	VCC3P3+0.3	V	Note 8
VIL3	Input Low Voltage	–0.3	0.8	V	Note 8
VIH3	Input High Voltage	2.0	VCCSUS3P3+0.3	V	Note 8
VIL4	Input Low Voltage	–0.1	0.35	V	
VIH4	Input High Voltage	0.75	VCCEPAUX+0.1	V	Note 8
VIL5	Input Low Voltage	-0.1	0.35	V	
VIH5	Input High Voltage	0.75	VCCSUS+0.1	V	
VIL6	Input Low Voltage	-0.1	0.35	V	
VIH6	Input High Voltage	0.75	VCCPCPU+0.1	V	
VIMIN7	Minimum Input Voltage	175	—	mVdiff p-p	Note 4
VIMAX7	Maximum Input Voltage	—	1200	mVdiff p-p	Note 4
VIL8	Input Low Voltage	–0.5	0.10	V	
VIH8	Input High Voltage	0.40	1.2	V	
VIMIN9 - Gen1i	Minimum Input Voltage - 1.5 Gb/s internal SATA	325	—	mVdiff p-p	Note 5
VIMAX9 -Gen1i	Maximum Input Voltage - 1.5 Gb/s internal SATA	—	600	mVdiff p-p	Note 5
VIMIN9 - Gen2i	Minimum Input Voltage - 3.0 Gb/s internal SATA	275	—	mVdiff p-p	Note 5
VIMAX9 -Gen2i	Maximum Input Voltage - 3.0 Gb/s internal SATA	—	750	mVdiff p-p	Note 5
Rin-Diff10	Received Differential Input Impedance GbE	80	120	Ω	
VIL11 (Absolute Minimum)	Input Low Voltage	-0.3	—	V	
VIH11 (Absolute Maximum)	Input High Voltage	—	1.150	V	
VIL12	Input Low Voltage	–0.5	0.78	V	
VIH12	Input High Voltage	2.0	VCC3P3_RTC+0.5	V	Note 6
VIL13	Input Low Voltage	–0.5	0.78	V	
VIH13	Input High Voltage	2.3	VCC3P3_RTC+0.5	V	Note 6
VIL_PECI	Input Low Voltage	-0.15	0.275(VCCPCPU)	V	
VIH_PECI	Input High Voltage	0.725(VCCPCPU)	VCCPCPU+0.15	V	

Table 33-7. DC Input Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Notes
VDI	Differential Input Sensitivity	0.2	—	V	Note 1,3
VCM	Differential Common Mode Range	0.8	2.5	V	Note 2,3
VSE	Single-Ended Receiver Threshold	0.8	2.0	V	Note 3
VHSSQ	HS Squelch Detection Threshold	100	150	mV	Note 7
VHSDSC	HS Disconnect Detection Threshold	525	625	mV	Note 7
VHSCM	HS Data Signaling Common Mode Voltage Range	–50	500	mV	Note 7

Notes:
1. V_{DI} = | USB[x]D[p] – USB[x]D[n]
2. Includes VDI range
3. Applies to Low-Speed/High-Speed USB
4. PCI Express mVdiff p-p = 2*|PCIE_EP_RX[p][x] – PCIE_EP_RX[n][x]|
5. SATA Vdiff, RX (VIMAX9/MIN9) is measured at the SATA connector on the receiver side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]RXp – SATA[x]RXn|
6. VCC3P3_RTC is the voltage applied to the VCC3P3_RTC well of the PCH. When the system is in a G3 state, this is generally supplied by the coin cell battery, but for S5 and greater, this is generally VCCSUS3P3.
7. Applies to High-Speed USB 2.0.
8. 3.3V refers to VCCSUS3P3 for signals in the suspend well and to VCC3P3 for signals in the core well and to VCC3P3 for signals in the ME well. See Section 32.0 for signal and power well association.

Table 33-8. DC Output Characteristic Signal Association (Sheet 1 of 2)

Symbol	Associated Signals
VOH1/VOL1	**Processor Signal:** PMSYNC; CPUPWRGD
VOH2/VOL2	**SPI EEPROM:** GBE_EE_CS#, GBE_EE_DI, GBE_EE_SK **GbE**: GBE[3:0]_SWDP[1:0], GBE[3:0]_LED
VOL3	**GbE**: GBE_WAKE# **I2C GbE Signals:** SFP[3:0]_I2C_CLK, SFP[3:0]_I2C_DATA **SMBus GbE Signals:** GBE_SMBCLK, GBE_SMBDAT, GBE_SMBALRT# **SPI EEPROM:** GBE_EE_CS#, GBE_EE_DI, GBE_EE_SK **SMBus Signals:** EP_SMBCLK, EP_SMBDAT, EP_SMBALRT#
VOH4/VOL4	**LPC/Firmware Hub Signals:** LAD[3:0], LFRAME#, INIT3_3V# **UART Signals:** SIU[1:0]_TXD, SIU[1:0]_RTS#, SIU[1:0]_DTR#, WDT_TOUT# **SATA Signals**: SATA_LED#, SCLOCK, SLOAD, SDATAOUT0, SDATAOUT1 **GPIO Signals:** GPIO [55, 54, 53, 52, 51, 50, 49, 48, 39, 38, 37, 36, 35, 34, 33, 32, 23, 22, 21, 20, 19, 18, 17,16, 7, 6, 5, 4, 3, 2, 1, 0] **SPI Signals:** SPI_CS0#, SPI_CS1#, SPI_MOSI, SPI_CLK **Interrupt Signals:** SERIRQ **Miscellaneous:** BBS0
VOH5/VOL5	**Power Management Signals:** SLP_S3#, SLP_S4#, SLP_S5#, SUS_CLK, SUS_STAT#, DRAMPWRGD **System Management Link Signals:** SML1CLK, SML1DAT, SML1ALERT# **GPIO Signals:** GPIO [75, 74, 73, 72, 63, 62, 61, 60, 59, 58, 57, 56, 47, 46, 45, 44, 43, 42, 41, 40, 31, 30, 28, 27, 26, 25, 24, 15, 14, 13, 12, 11, 10, 9, 8]
VOL6	**SMBus Signals:** MST_SMBCLK, MST_SMBDAT, MST_SMBALERT# (open drain suspend signals)
VOL7	**JTAG Signals:** JTDO
VOL8	**JTAG Signals:** EP_JTDO

Table 33-8. DC Output Characteristic Signal Association (Sheet 2 of 2)

Symbol	Associated Signals
VOH9/VOL9	**Reset Signals:** PLTRST#
VOH10/VOL10	**USB Signals**: USB[5:0]D[p,n] in Low-speed and Full-speed Modes
VOMIN11/VOMAX11	**SATA Data TX Signals:** SATA[5:4]_TX[P,N]
VOMIN12/VOMAX12	**PCI Express* Data TX Signals:** PCIE_EP_TX[p,n][15:0], PCIE_RC_TX[p,n][3:0], DMI_TX[p,n][3:0]
V_{OMIN13}/V_{OMAX13}	**GbE:** SRDSO_[3:0]_P, SRDSO_[3:0]_N
VOH_PECI/VOL_PECI	**PECI Signal:** PECI
VHSOI VHSOH VHSOL VCHIRPJ VCHIRPK	**USB Signals:** USB[5:0]D[p,n] in High-speed Mode

Table 33-9. DC Output Characteristics (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	I_{OL} / I_{OH} Max	Notes
V_{OL1}	Output Low Voltage	—	0.255	V	3 mA	
V_{OH1}	Output High Voltage	VCCPCPU - 0.3	VCCPCPU	V	-3 mA	
V_{OL2}	Output Low Voltage	—	0.4	V	10 mA	
V_{OL2}	Output Low Voltage	—	0.2	V	100 uA	
V_{OH2}	Output High Voltage	2.4	—	V	-10 mA	
V_{OH2}	Output High Voltage	VCCEP3P3AUX-0.2	—	V	-100 uA	
V_{OL3}	Output Low Voltage	—	0.4	V	10 mA	Note 1,4
V_{OL3}	Output Low Voltage	—	0.2	V	100 uA	Note 1,4
V_{OL4}	Output Low Voltage	—	0.4	V	6 mA	Note 1
V_{OH4}	Output High Voltage	VCC3P3- 0.5	VCC3P3	V	-2 mA	Note 1
V_{OL5}	Output Low Voltage	—	0.4	V	6 mA	
V_{OH5}	Output High Voltage	VCCSUS3P3- 0.5	VCCSUS3P3	V	-2 mA	
V_{OL6}	Output Low Voltage	—	0.4	V	4 mA	Note 1, 5
V_{OL7}	Output Low Voltage	—	VCCEPAUX/2	V	12 mA	Note 1
V_{OL8}	Output Low Voltage	—	VCCSUS/2	V	12 mA	
V_{OL9}	Output Low Voltage	—	0.1 (VCCSUS3P3)	V	1.5 mA	
V_{OH9}	Output High Voltage	0.9 (VCCSUS3P3)	VCCSUS3P3	V	-2.0 mA	
V_{OL10}	Output Low Voltage	—	0.4	V	6 mA	
V_{OH10}	Output High Voltage	VCCSUS3P3_USB-0.5	VCCSUS3P3_USB	V	-2 mA	
$V_{OMIN11Gen1i,m}$	Minimum Output Voltage	400	—	mVdiff p-p		Note 3

Table 33-9. DC Output Characteristics (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	I_{OL} / I_{OH} Max	Notes
$V_{OMAX11Gen1i,m}$	Maximum Output Voltage	—	600	mVdiff p-p		Note 3
$V_{OMIN11Gen2i,m}$	Minimum Output Voltage	400	—	mVdiff p-p		Note 3
$V_{OMAX11Gen2i,m}$	Maximum Output Voltage	—	700	mVdiff p-p		Note 3
V_{OMIN12}	Output Low Voltage	400	—	mVdiff p-p		Note 2
V_{OMAX12}	Output High Voltage	—	600	mVdiff p-p		Note 2
V_{OMIN13} SerDes/SGMII	Output Low Voltage	800	—	mVdiff p-p		
V_{OMAX13} SerDes/SGMII	Output High Voltage	—	1200	mVdiff p-p		
VOL_PECI	Output Low Voltage		0.25 (VCCPCPU)	V	0.5 mA	
VOH_PECI	Output High Voltage	0.75 (VCCPCPU)	VCCPCPU	V	-6 mA	
VHSOI	HS Idle Level	–10.0	10.0	mV		
VHSOH	HS Data Signaling High	360	440	mV		
VHSOL	HS Data Signaling Low	–10.0	10.0	mV		
VCHIRPJ	Chirp J Level	700	1100	mV		
VCHIRPK	Chirp K Level	–900	–500	mV		

Notes:
1. The GBE_SMBDAT, GBE_SMBCLK, GBE_SMBALRT#, EP_SMBDAT, EP_SMBCLK, EP_SMBALRT#, MST_SMBDAT, MST_SMBCLK, MST_SMBALRT#, JTDO, and EP_JTDO have open drain driver, and the VOH does not Apply. SATA_LED# has an open-collector driver, and the VOH spec does not apply. These signals must have external pull up resistor.
2. PCI Express mVdiff p-p = 2*|PCIE_EP_T[p][x]– PCIE_EP_T[n][x]|, PCI Express mVdiff p-p = 2*|PCIE_RC_T[p][x] – PCIE_RC_T[n][x]|
3. SATA Vdiff, tx (VOMIN7/VOMAX7) is measured at the SATA connector on the transmit side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]TXP – SATA[x]TXN|
4. Reference supply or power well VCCEP3P3AUX.
5. Reference supply or power well VCCSUS3P3.

33.6 PCI Express* Root Complex and DMI Gen1 DC/AC Characteristics

Table 33-10. PCI Express* and DMI Gen1 TX Specification Interface Timings (Sheet 1 of 2)

Symbol	Parameter	Min	Nom	Max	Unit	Notes
UI	Unit Interval	399.88	400	400.12	ps	
$V_{TX-DIFFpp}$	Differential Peak to Peak Output Voltage	0.800		1.2	V	Note 1
$T_{TX-DE-RATIO}$	De-Emphasized Differential Output Voltage (Ration)	-3.0	-3.5	-4.0	dB	
T_{TX-EYE}	Minimum TX Eye Width	0.75			UI	

Table 33-10. PCI Express* and DMI Gen1 TX Specification Interface Timings (Sheet 2 of 2)

Symbol	Parameter	Min	Nom	Max	Unit	Notes
$T_{TX-EYE-MEDIAN-to-MAX-JITER}$	Maximum time between the jitter median and maximum deviation from the median.			0.125	UI	
$T_{TX-RISE}$, $T_{TX-FALL}$	D+/D- TX Output Rise/Fall Time	0.125			UI	
$V_{TX-CM-ACp}$	RMS AC Peak Common Mode Output Voltage			20	mV	
$V_{TX-CM-DC-LINE-DELTA}$	Absolute Delta of DC Common Mode Voltage between D+ and D-	0		25	mV	
$V_{TX-DC-CM}$	The TX DC Common Mode Voltage	0		3.6	V	
$I_{TX-SHORT}$	TX Short Circuit Current Limit			90	mA	
$RL_{TX-DIFF}$	Differential Return Loss	10			dB	
$Z_{TX-DIFF-DC}$	DC Differential TX Impedance	80	100	120	Ω	

1. DMI DC coupling uses half signal level (1/2 of the MIN and MAX)

Note: For additional information, see the *PCI Express Protocol, Rev1.1*

Figure 33-1. PCI Express Gen1 Transmitter Eye

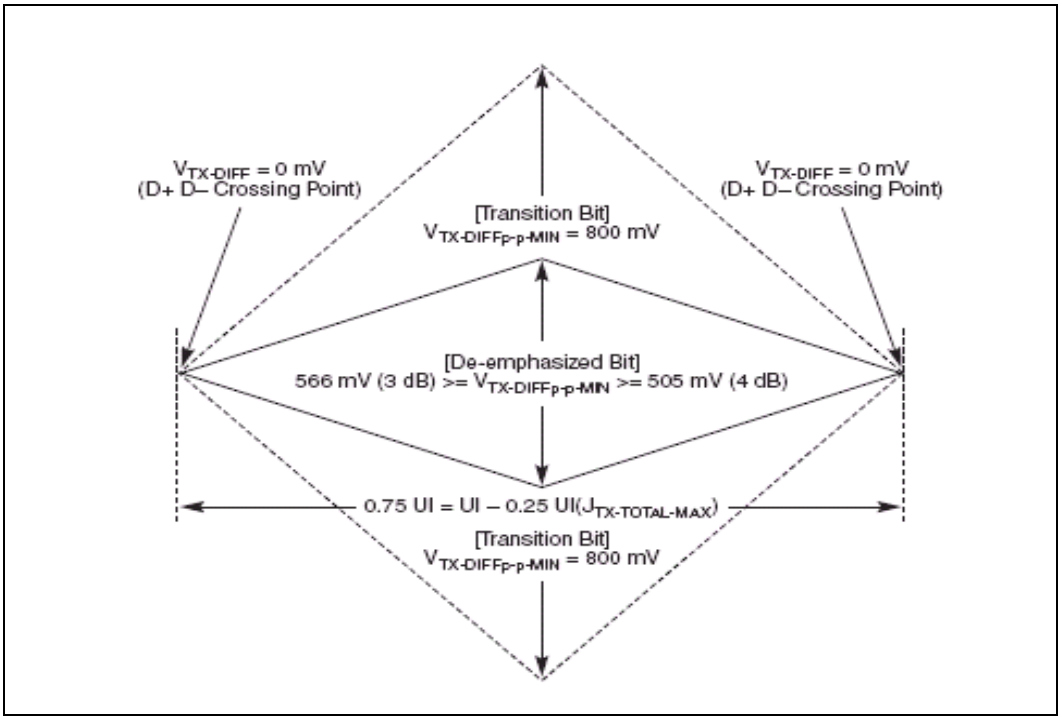


Table 33-11. PCI Express* and DMI Gen1 RX Specification Interface Timings (Sheet 1 of 2)

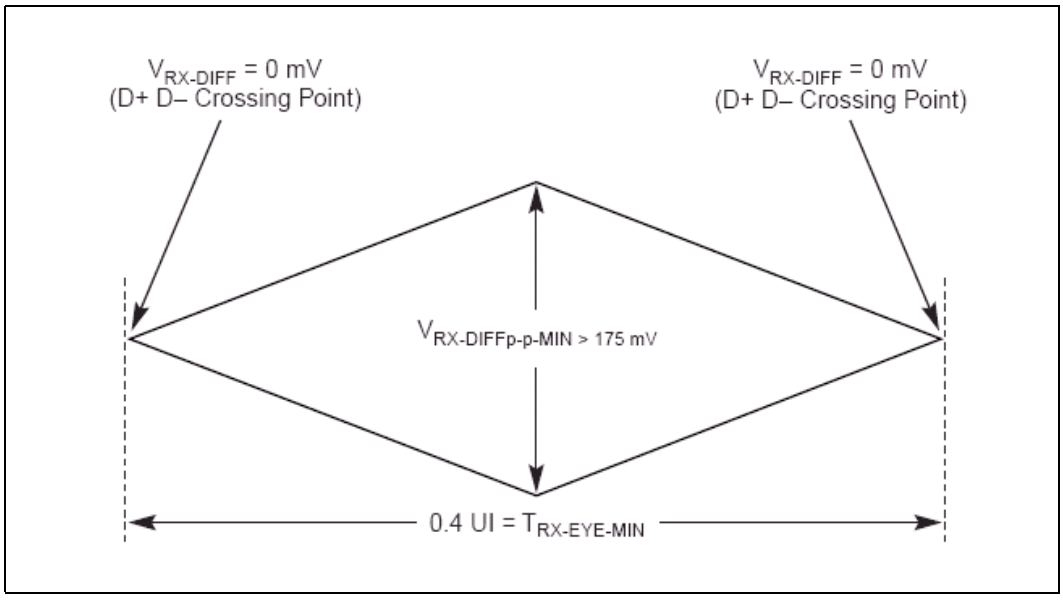
Symbol	Parameter	Min	Nom	Max	Unit
UI	Unit Interval	399.88	400	400.12	ps
$V_{RX-DIFFpp}$	Differential Input Peak to Peak Voltage	0.175		1.2	V
T_{RX-EYE}	Minimum Receiver Eye Width	0.4			UI

Table 33-11. PCI Express* and DMI Gen1 RX Specification Interface Timings (Sheet 2 of 2)

Symbol	Parameter	Min	Nom	Max	Unit
$T_{RX-EYE-MEDIAN-to-MAX-JITER}$	Maximum time between the jitter median and maximum deviation from the median.			0.3	UI
$V_{RX-CM-ACp}$	AC Peak Common Mode Input Voltage			150	mV
$RL_{RX-DIFF}$	Differential Return Loss	10			dB
RL_{RX-CM}	Common Mode Return Loss	6			dB
$Z_{RX-DIFF-DC}$	DC Differential Input Impedance	80	100	120	Ω
Z_{RX-DC}	DC Input Impedance	40	50	60	Ω

Note: For additional information, see the *PCI Express Protocol*, Rev1.1

Figure 33-2. PCI Express Gen1 Receiver Eye



33.7 PCIe* Gen2 EndPoint DC/AC Specification

Table 33-12. PCI Express* Gen2 TX Specification Interface Timings

Symbol	Parameter	2.5 GT/S	5.0 GT/S	Unit
UI	Unit Interval	399.88(min) 400.12(max)	199.94(min) 200.06(max)	ps
$V_{TX-DIFF-PP}$	Differential Peak to Peak Output Voltage	0.8(min) 1.2(max)	0.8(min) 1.2(max)	V
$V_{TX-DIFF-PP-LOW}$	Low Power Differential Peak to Peak Output Voltage Swing	0.4(min) 1.2(max)	0.4(min) 1.2(max)	V
$V_{TX-DE-RATIO-3.5dB}$	De-Emphasized Differential Output Voltage (Ration)	3.0(min) 4.0(max)	3.0(min) 4.0(max)	dB
$V_{TX-DE-RATIO-6dB}$	De-Emphasized Differential Output Voltage (Ration)	N/A	5.5(min) 6.5(max)	dB
$T_{MIN-PULSE}$	Instantaneous LonePulse Width	Not Specified	0.9(min)	UI
T_{TX-EYE}	Minimum TX Eye Width	0.75(min)	0.75(min)	UI
$T_{TX-EYE-MEDIAN-to-MAX-JITER}$	Maximum time between the jitter median and maximum deviation from the median.	0.125(max)	Not Specified	UI
$T_{TX-HF-DJ-DD}$	TX Deterministic Jitter >1.5 MHz	Not Specified	0.15 (max)	UI
$T_{TX-LF-RMS}$	TX RMS Jitter <1.5 MHz	Not Specified	3.0	ps RMS
$T_{TX-RISE}$, $T_{TX-FALL}$	TX Output Rise/Fall Time	0.125 (min)	0.15 (max)	UI
$T_{RF-MISMATCH}$	TX Output Rise/Fall Mismatch	Not Specified	0.1 (max)	UI
BW_{TX-PLL}	Maximum TX PLL Bandwidth	22 (max)	16 (max)	MHz
$BW_{TX-PLL-LO-3dB}$	Minimum TX PLL Bandwidth for 3dB Peaking	1.5 (min)	8 (min)	MHz
$BW_{TX-PLL-LO-1dB}$	Minimum TX PLL Bandwidth for 1dB Peaking	Not Specified	5 (min)	MHz
$PKG_{TX-PLL1}$	TX PLL Peaking with 8MHz min BW	Not Specified	3.0 (max)	dB
$PKG_{TX-PLL2}$	TX PLL Peaking with 5MHz min BW	Not Specified	1.0 (max)	dB
$RL_{TX-DIFF}$	TX Package plus Si Differential Return Loss	10 (min)	10 (min) for 0.05-1.25 GHz 8 (min) for 1.25-2.5 GHz	dB
RL_{TX-CM}	TX Package plus Si Differential Common Mode Return Loss	6 (min)	6 (min)	dB
$Z_{TX-DIFF-DC}$	DC Differential TX Impedance	80 (min) 120 (max)	120 (max)	Ω
$V_{TX-CM-AC-PP}$	TX AC Peak Common Mode Output Voltage (5.0 GT/s)	Not Specified	100 (max)	mVpp
$V_{TX-CM-AC-PP}$	TX AC Peak Common Mode Output Voltage (2.5 GT/s)	Not Specified	100 (max)	mVpp
$I_{TX-SHORT}$	TX Short Circuit Current Limit	90(max)	90 (max)	mA
$V_{TX-DC-CM}$	The TX DC Common Mode Voltage	0 (min) 3.6 (max)	0 (min) 3.6 (max)	V
$V_{TX-CM-DC-LINE-DELTA}$	Absolute Delta of DC Common Mode Voltage between D+ and D-	0 (min) 25 (max)	0 (min) 25 (max)	mV
C_{TX}	AC Coupling Capacitor	75 (min)[1] 200 (max)	75 (min)[1] 200 (max)	nF

1. For additional information, see the *PCI Express Protocol*, Rev2.0.

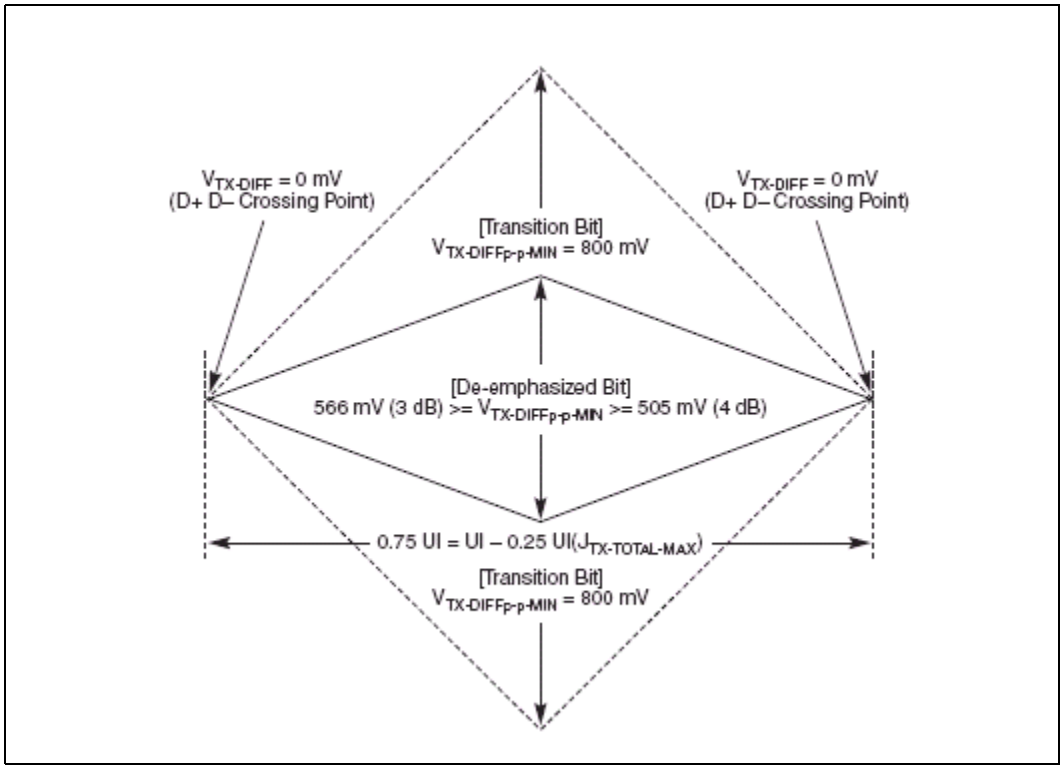


Figure 33-3. PCI Express Transmitter Eye

Table 33-13. PCI Express* Gen2 RX Specification Interface Timings (Sheet 1 of 2)

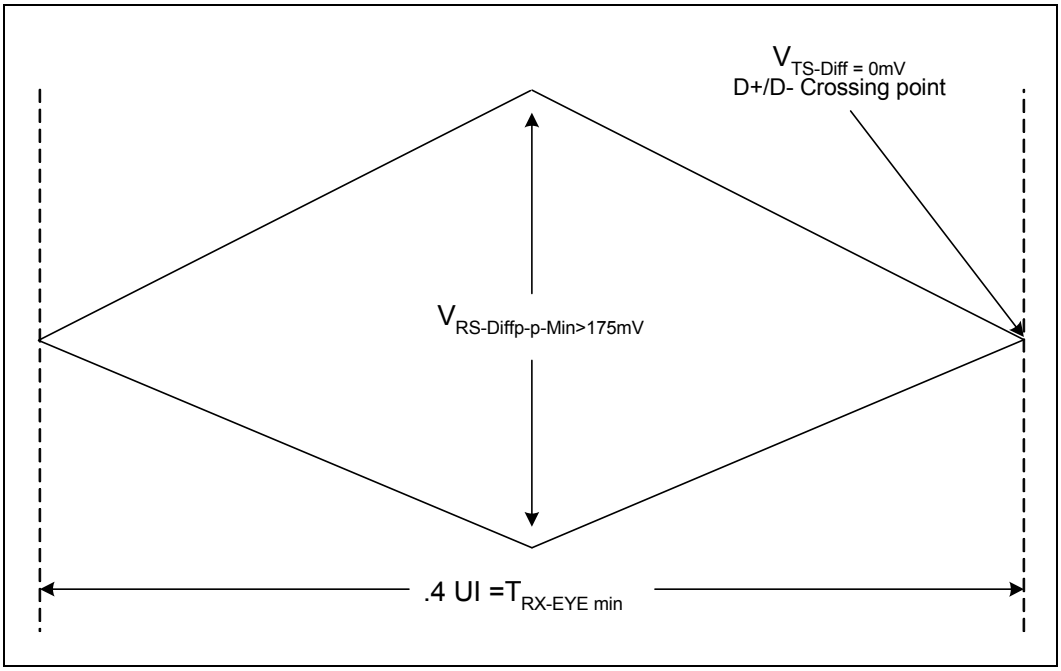
Symbol	Parameter	2.5 GT/S	5.0 GT/S	Unit
UI	Unit Interval	399.88 (min) 400.12 (max)	199.94 (min) 200.06 (max)	ps
$V_{RX-DIFF-PP-CC}$	Differential Input Peak to Peak Voltage for Common Refclk RX Architecture	0.175 (min) 1.2 (max)	0.120 (min) 1.2 (max)	V
T_{RX-EYE}	Minimum Receiver Eye Width	N/A	0.40 (min)	UI
$T_{RX-TJ-CC}$	Max Receiver Inherent Timing Error	N/A	0.40 (max)	UI
$T_{RX-DJ-DD-CC}$	Max Receiver Inherent Deterministic Timing Error	N/A	0.30 (max)	UI
$T_{RX-EYE-MEDIAN-to-MAX-JITER}$	Maximum time between the jitter median and maximum deviation from the median.	0.3 (max)	Not Specified	UI
$T_{RX-MIN-PULSE}$	Minimum Width Pulse at RX	Not Specified	0.6 (min)	UI
$V_{RX-MAX-MIN-RATIO}$	Min/Max Pulse Voltage on Consecutive UI	Not Specified	5 (max)	-
$BW_{RX-PLL-HI}$	Maximum RX PLL Bandwidth	22 (max)	16 (max)	MHz
$BW_{RX-PLL-LO-3dB}$	Minimum RX PLL Bandwidth for 3dB peaking	1.5 (min)	8 (min)	MHz
$BW_{RX-PLL-LO-1dB}$	Minimum RX PLL Bandwidth for 1dB peaking	Not Specified	5 (min)	MHz
$PKG_{RX-PLL1}$	RX PLL peaking with 8 MHz minimum Bandwidth	Not Specified	3.0	dB

Table 33-13. PCI Express* Gen2 RX Specification Interface Timings (Sheet 2 of 2)

Symbol	Parameter	2.5 GT/S	5.0 GT/S	Unit
$PKG_{RX-PLL2}$	RX PLL peaking with 5 MHz minimum Bandwidth	Not Specified	1.0	dB
$RL_{RX-DIFF}$	Rx Package plus Si Differential Return Loss	10 (min)	10 (min) for 0.05-1.5GHz 8 (min) for 1.25-2.5GHz	dB
RL_{RX-CM}	Common Mode Return Loss	6 (min)	6 (min)	dB
$Z_{RX-DIFF-DC}$	DC Differential Input Impedance	80 (min) 120 (max)	Not Specified	Ω
Z_{RX-DC}	DC Input Impedance	40 (min) 60 (max)	40 (min) 60 (max)	Ω
$v_{RX-CM-AC-P}$	RX AC Common Mode Voltage	150 (max)	150 (max)	mVP

Note: For additional information, see the *PCI Express Protocol*, Rev 2.0.

Figure 33-4. PCI Express Receiver Eye



33.7.1 PCIe* Specification - Input Clock

The input clock for PCIe* must be a differential input clock in frequency of 100 MHz. For full specifications, see the *PCI-Express Card Electromechanical Specifications* (refclk specifications).

33.8 SATA Gen1/Gen2 DC/AC Characteristics

Table 33-14. SATA Specification Interface Timings (Sheet 1 of 2)

Symbol	Parameter	Gen1		Gen2		Units	Notes
		Min	Max	Min	Max		
UI	Unit Interval	666.43	670.23	333.21	335.11	ps	
Receiver Parameter							
Z_{diffRX}	RX Pair Differential Impedance	85	115	85	115	Ω	
$T_{J\text{-}Con\text{-}DD\text{-}5}$	TJ at Connector, Data-Data, 5		0.43			UI	
$D_{J\text{-}Con\text{-}DD\text{-}5}$	DJ at Connector, Data-Data, 5		0.35			UI	
$T_{J\text{-}Con\text{-}DD\text{-}250}$	TJ at Connector, Data-Data, 250		0.60			UI	
$D_{J\text{-}Con\text{-}DD\text{-}250}$	DJ at Connector, Data-Data, 250		0.42			UI	
$T_{TJ\text{-}Con\text{-}CD\text{-}10}$	TJ at Connector, Clk-Data, fBAUD/10				0.46	UI	
$T_{DJ\text{-}Con\text{-}CD\text{-}10}$	DJ at Connector, Clk-Data, fBAUD/10				0.35	UI	
$T_{TJ\text{-}Con\text{-}CD\text{-}500}$	TJ at Connector, Clk-Data, fBAUD/500				0.60	UI	
$T_{DJ\text{-}Con\text{-}CD\text{-}500}$	DJ at Connector, Clk-Data, fBAUD/500				0.42	UI	
$T_{jcon\ CD\text{-}1667}$	Tj at Connector Clk-Data, fbaud/1667		.65		.65	UI	
$D_{j\ con\ CD\text{-}1667}$	Dj at Connector Clk-Data, fbaud/1667		.35		.35	UI	
$RL_{DD11,RX}$	Min Differential Mode return loss (limits 150 - 300 MHz)			18		dB	
	Min Differential Mode return loss (limits 1.2 - 2.4 GHz)			8		dB	
	Min Differential Mode return loss (limits 2.4 - 3.0 GHz)			3		dB	
$RL_{CC11,RX}$	RX Common Mode Return Loss (limits 150 - 600 MHz)			5		dB	
	RX Common Mode Return Loss (limits 1.2 - 2.4 GHz)			2		dB	
	RX Common Mode Return Loss (limits 3.0 - 5.0 GHz)			1		dB	
V_{diffRX}	RX Differential Input Voltage	240	600	240	750	mV	
Transmitter Parameter							
Z_{diffTX}	RX Pair Differential Impedance	85	115	85	115	Ω	
$v_{TX\text{-}DIFF\text{-}PP}$	VdiffTX, TX Differential Input Voltage	400	600	400	700	mvPP	
$T_{TX\text{-}RISE}$	TX Rise Time	100	273	67	136	ps	Note 1
$T_{TX\text{-}FALL}$	TX Fall Time	100	273	67	136	ps	Note 1
$T_{TX\text{-}SKEW}$	TX Differential Skew	-	20	-	20	ps	
$V_{CM\text{-}AC}$	TX AC Common Mode				50	mv	
$D_{J\text{-}Con\text{-}DD\text{-}250}$	DJ at Connector, Data-Data, 250		0.220			UI	
$T_{TJ\text{-}Con\text{-}CD\text{-}500}$	TJ at Connector, Clk-Data, fBAUD/500				0.37	UI	
$T_{DJ\text{-}Con\text{-}CD\text{-}500}$	DJ at Connector, Clk-Data, fBAUD/500				0.19	UI	
$T_{jcon\ CD\text{-}1667}$	Tj at Connector Clk-Data, fbaud/1667		.65		.65	UI	
$D_{j\ con\ CD\text{-}1667}$	Dj at Connector Clk-Data, fbaud/1667		.35		.35	UI	

Table 33-14. SATA Specification Interface Timings (Sheet 2 of 2)

Symbol	Parameter	Gen1		Gen2	Units	Notes
$RL_{DD11,TX}$	Min Differential Mode return loss (limits 150 - 300 MHz)			14	dB	
	Min Differential Mode return loss (limits 1.2 - 2.4 GHz)			6	dB	
	Min Differential Mode return loss (limits 2.4 - 3.0 GHz)			3	dB	
$RL_{CC11,TX}$	RX Common Mode Return Loss (limits 150 - 600 MHz)			5 - 8	dB	
	RX Common Mode Return Loss (limits 1.2 - 2.4 GHz)			2	dB	
	RX Common Mode Return Loss (limits 3.0 - 5.0 GHz)			1	dB	

1. 20% - 80%

Note: All parameters measured at Rload = 100Ω ±10% load.

- For a detailed description of the symbols, see the *IEEE1596.3-1996* Standard.

33.9 Gigabit Ethernet SGMII DC/AC Characteristics

Table 33-15. GbE SGMII Driver DC Specification Interface

Symbol[1]	Parameter[2]	Min	Nom	Max	Unit
$v_{Out-High}$	Output voltage high			1200	mV
$v_{Out-Low}$	Output voltage low	800			mV
v_{Ring}	Output ringing			10	%
$\|v_{Out-Diff}\|$	Output Differential Voltage	400		600	mV
R_{Out}	Output impedance (single ended)	40		60	Ω
$\|v_{Out-Diff}\|$	Change in $v_{Out-Diff}$ between "0" and "1"			25	mV
IShort-a, IShort-b	Output current on Short to GND		40		mA

1. For a detailed description of the symbols, see the IEEE1596.3-1996 standard
2. All parameters measured at Rload = 100Ω ±10% load

Table 33-16. GbE SGMII Receiver DC Specification Interface (AC Coupled)

Symbol[1]	Parameter[2]	Min	Max	Unit
V_I	Input voltage range a or b	675	1725	mV
Vidth	Input differential threshold	-50	+50	mV
	Input differential hystersis	25		mV
$R_{In-Diff}$	Received differential input impedance	80	120	Ω

1. For a detailed description of the symbols see the IEEE1596.3-1996 standard
2. All parameters measured at Rload = 100Ω ±10% load

Table 33-17. GbE SGMII Driver AC Specification Interface

Symbol[1]	Parameter	Min	Nom	Max	Unit
	Signal Speed, per lane		1.25		Gb/Sec
T_{Fall}	$v_{Out-Diff}$ Fall time (20%-80%)		130		pSec
T_{Rise}	$v_{Out-Diff}$ Rise time (20%-80%)		130		pSec

1. For a detailed description of the symbols, see the IEEE1596.3-1996 standard

33.10 SerDes DC/AC Specification

The SerDes interface supports the following standards:

1. PICMG 3.1 specification Rev 1.0 1000BASE-BX.
2. 1000BASE-KX electrical specification defined IEEE802.3ap clause 70.
3. SGMII on 1000BASE-BX or 1000BASE-KX compliant electrical interface (AC coupling with internal clock recovery).
4. SFP (Small Form-Factor Pluggable) Transceiver Rev 1.0

The specifications define the interface for the back-plane board connection, interface to external 1000BASE-T PHY and the interface to fiber or SFP module.

Table 33-18. SerDes GbE 1000 Base RX Specifications

Symbol	Parameter	Min	Nom	Max	Unit
	Signal Speed, per lane		1.25		Gb/Sec
BER	Target Bit Error Rate		10^{-12}		
$V_{RX-Diffpp}$	Differential Peak to Peak Input Voltage			1600	mV

Table 33-19. SerDes GbE 1000 Base TX Specifications

Symbol	Parameter	Min	Nom	Max	Unit
	Signal Speed, per lane		1.25		Gb/Sec
$V_{TX-Diffpp}$	Differential Peak to Peak Output Voltage	800		1200	mV
$V_{TX-Diffpp-DISABLE}$	Differential Peak to Peak Output Voltage TX Disabled			30	mV
V_{TX-CM}	Common Mode Voltage Limits	-0.15		1.2	V
V_{TX}-DIFF TEMP	Differential Output Template				
TTRAN-TIME	Transition Time (20%-80%)			130	PS
T_{TX}-RANDOM-JITTER	Random Jitter Peak to Peak			0.27	UI
T_{TX}-DETERMIN-JITTER	Deterministic Jitter Peak to Peak			0.17	UI
T_{TX}-TOTAL-JITTER	Total Jitter Peak to Peak			0.35	UI

33.11 CRU Clock DC/AC Specification

Table 33-20. CRU Differential Input Clock DC Specifications

Symbol	Parameter	Min	Typical	Max	Units	Figure	Notes
CRU Clock (CRU_CLK100[P,N]) from a clock chip							1
V_{IL}	Voltage Input Low	-0.150	0.00	0.150	V	33-5	1
V_{IH}	Voltage Input High	0.660	0.710	0.850	V	33-5	1
V_{ISE}	Voltage Input Single Ended	-0.1		1.15	V	33-5	1
$V_{CROSS(abs)}$	Absolute Crossing Point	0.250	N/A	0.550	V	33-5, 33-6	1, 2, 3, 6, 9
$V_{CROSS(rel)}$	Relative Crossing Point	0.250+0.5 (VH_{avg}-0.71)	N/A	0.550+0.5 (VH_{avg}-0.71)	V	33-5, 33-6	1, 2, 3, 6, 7, 9
ΔV_{CROSS}	Range of Crossing Point	N/A	N/A	0.140	V	33-5, 33-6	1, 2, 8, 9
V_{RBM}	Ringback Margin	0.200	N/A	N/A	V	33-5	1, 4, 9
V_{TM}	Threshold Margin	V_{CROSS}-0.10	N/A	V_{CROSS}+0.10	V	33-5	1, 5, 9
V_{TM}	Threshold Margin	V_{CROSS}-0.10	N/A	V_{CROSS}+0.10	V	33-5	1, 5, 9

Notes:
1. Unless otherwise noted, all specifications in this table apply to all processor frequencies.
2. Crossing voltage is defined as the instantaneous voltage value when the rising edge of CRU_CLK100P equals the falling edge of CRU_CLK100N.
3. VHavg is the statistical average of the VH measured by the oscilloscope.
4. Ringback Margin is defined as the absolute voltage difference between the maximum Rising Edge Ringback and the maximum Falling Edge Ringback.
5. Threshold Region is defined as a region entered around the crossing point voltage in which the differential receiver switches. It includes input threshold hysteresis.
6. The crossing point must meet the absolute and relative crossing point specifications simultaneously.
7. VHavg can be measured directly using "Vtop" on Agilent* scopes and "High" on Tektronix* scopes.
8. ΔV_{CROSS} is defined as the total variation of all crossing voltages as defined in note 2.
9. Guaranteed by design. These values are typical values seen for this process, but not measured during production testing.

Figure 33-5. CRU Differential Clock Waveform

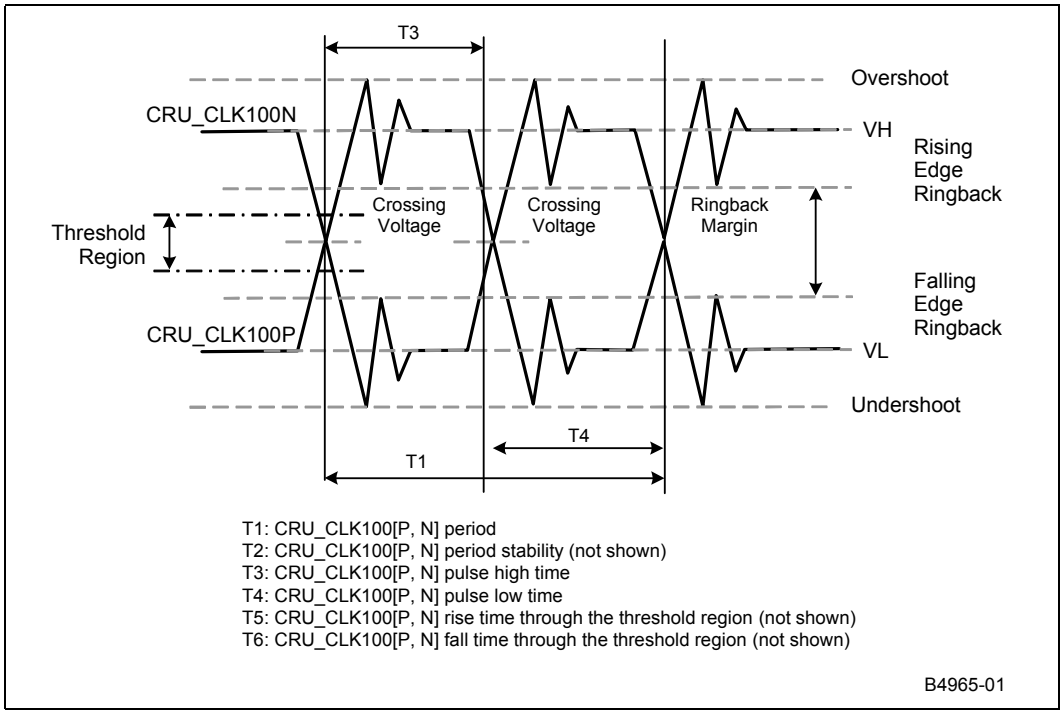


Figure 33-6. CRU Differential Clock Cross-Point Specification

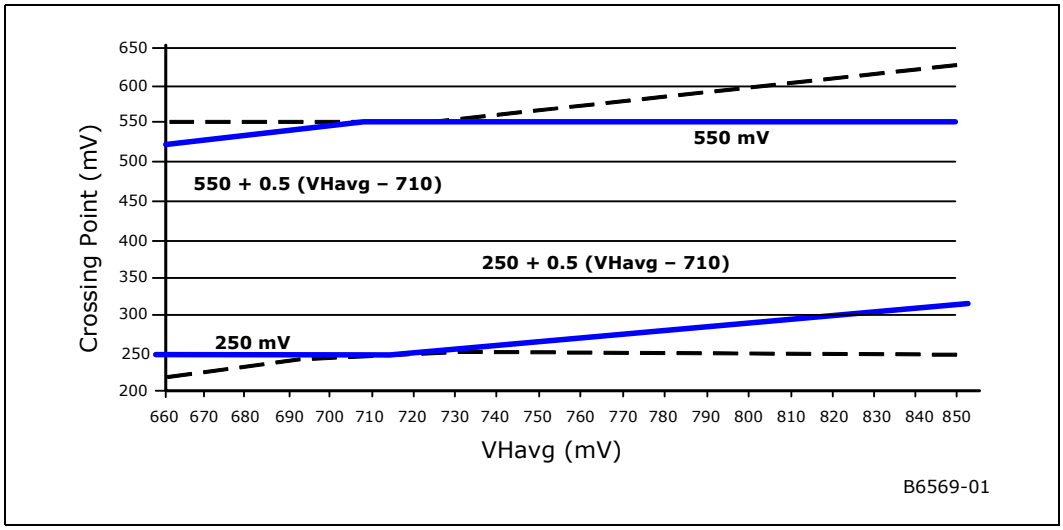


Table 33-21. CRU Differential Input Clock Timing Specifications

Symbol	Parameter	Min	Nom	Max	Unit	Figures	Notes
CRU Clock (CRU_CLK100[P,N]) from a clock chip							
F	CRU_CLK100 Frequency		-	100	MHz	-	-
Duty Cycle	Duty Cycle	45	-	55	%	-	-
T1	CRU_CLK100 Period	10.00	-	10.20	ns	33-5	1, 6
T2	CRU_CLK100 Period Stability	N/A	-	200.00	ps	-	2, 6
PPM	Parts per million Frequency Tolerance	-350		350	ppm	-	-
T_{jitter}	Cycle to cycle Jitter	-	50	150	ps	-	-
T3	T_{PH} CRU_CLK100 Pulse High Time	3.94	5.00	6.12	ns	33-5	4, 6
T4	T_{PL} CRU_CLK100 Pulse Low Time	3.94	5.00	6.12	ns	33-5	4, 6
T5	CRU_CLK100 Rise Time	175	-	700	ps	33-5	3, 5, 6
T6	CRU_CLK100 Fall Time	175	-	700	ps	33-5	3, 5, 6

Notes:
1. The period specified here is the average period. A given period may vary from this specification as governed by the period Stability specification (T2).
2. In this context, period stability is defined as the worst case timing difference between successive crossover voltages. In other words, the largest absolute difference between adjacent clock periods must be less than the period stability.
3. Slew rate is measured between the 35% and 65% points of the clock swing (VL to VH).
4. Combining the longest clock-high and clock-low times would violate the max clock period, and that combining the shortest clock-high and clock-low times would violate the minimum clock period. The clock-high and clock-low times specify the most extreme allowable combination of clock period and duty cycle.
5. Slew rate specifications apply to both rising and falling edges.
6. Guaranteed by design. These values are typical values seen for this process, but not measured during production testing.

33.12 SATA, DMI, and PCIe* Clocks DC/AC Specification

The specifications that follow are from the *CK505 Clock Synthesizer/Driver Specification*. More details can be obtained in the CK505 specification. These specifications are valid at the silicon pad only. Ledges on the clock signals will be observed if measuring the clocks at the vias on the bottom of the component, since the clocks rely on reflective switching. The Frequency and Period numbers in Table 33-22 are taken directly from the "AbsPerMin" and "AbsPerMax" clock-clock period variation in the *CK505 Clock Synthesizer/Driver Specification,* with SSC enabled (except for the display clock).

Table 33-22. SATA, DMI, and PCIe* Differential Input Clock DC Specification

Symbol	Parameter	Min	Nom	Max	Unit
V_{IL}	Voltage Input Low			-0.15	V
V_{IH}	Voltage Input High	0.15			V
V_{ISE}	Voltage Input Single Ended	-0.1		1.15	V
V_{Cross}	Absolute Crossing Point	0.25		0.55	V
$V_{Cross\text{-}delta}$	Vcross Variation			0.14	V
$V_{max\ (Absolute\ Overshoot)}$	Single-ended max voltage			1.15	V
$V_{min\ (Absolute\ Undershoot)}$	Single-ended min voltage	-0.1			V
$V_{rb\text{-}diff}$	Differential ringback voltage threshold	-0.10		.10	V

Table 33-23. SATA, DMI, and PCIe* Differential Input Clock AC Specification

Symbol	Parameter	Min	Nom	Max	Unit
SATA, DMI, and PCIe* Clocks (SATA_CLK100[P,N], DMI_CLK100[P,N], PCIE_EP_CLK100[P,N]) from a clock chip					
F	Frequency		100.0		MHz
$T_{period-abs}$	Absolute Period (including Jitter and Spread Spectrum)	9.847		10.203	ns
Duty Cycle	Duty Cycle of reference clock	40		60	%
PPM	Parts per million Frequency Tolerance	-300		300	%
ERRefclk-diffRrise ERRefclk-diffFall	Differential Rising and Falling edge rates	0.6		4	V/ns
T_{Stable}	Allowed time before ringback	500			ps
T_{jitter}	Cycle to cycle jitter	—		86	ps

Table 33-23, "SATA, DMI, and PCIe* Differential Input Clock AC Specification" includes specs for common clock platforms. For a design that has separate clock platforms, see the PCIe* Rev2 specification for the separate refclock architecture specifications.

Figure 33-7. SATA, DMI, and PCIe* Clocks Differential Clock Waveform

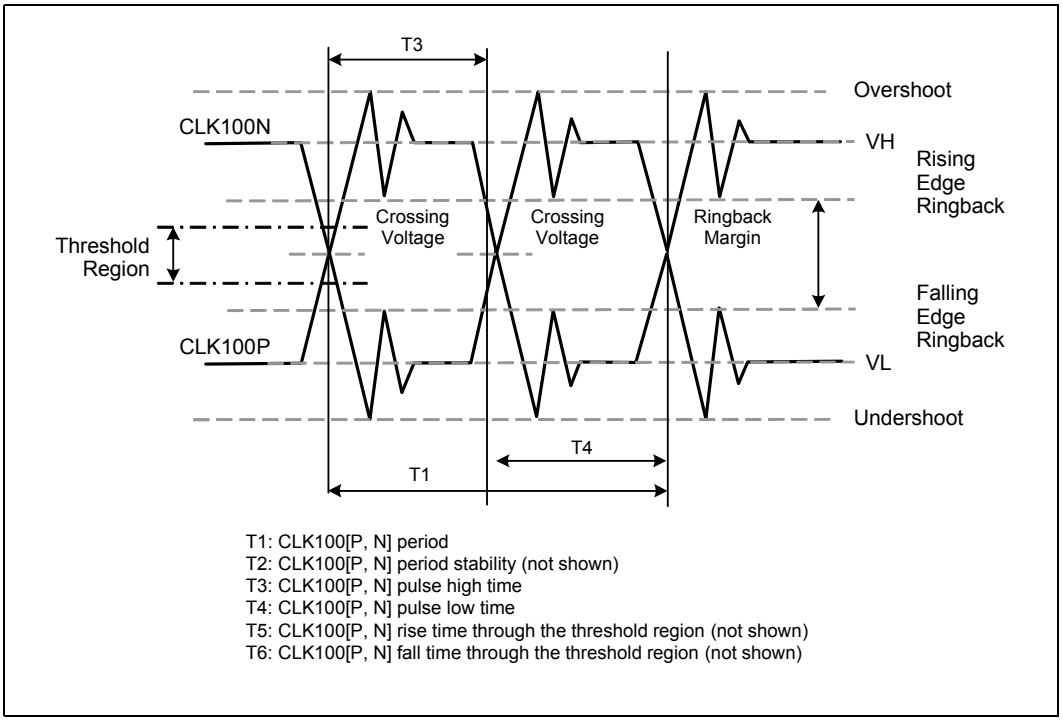


Figure 33-8. Single-Ended Measurement Point for Absolute Cross Point and Swing

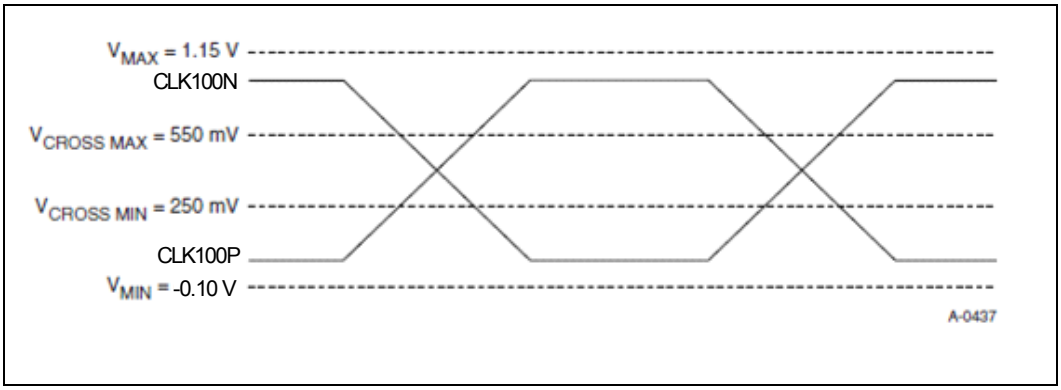


Figure 33-9. Single-Ended Clock Measurement Points for Delta Cross Point

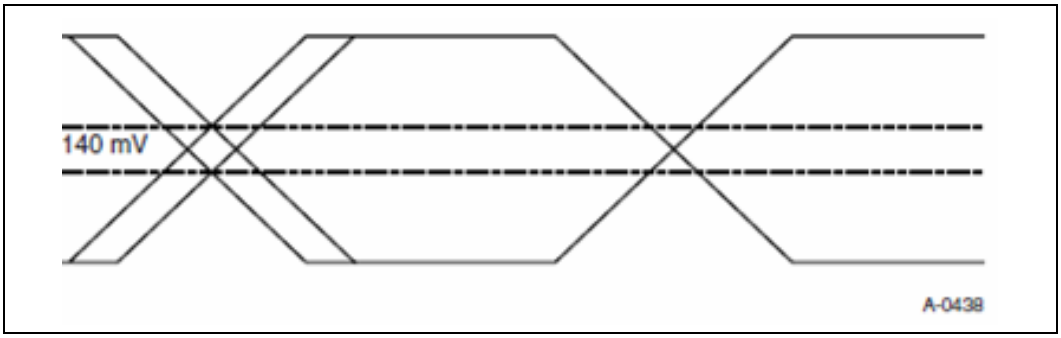


Figure 33-10.Differential Clock Cross Point Specification

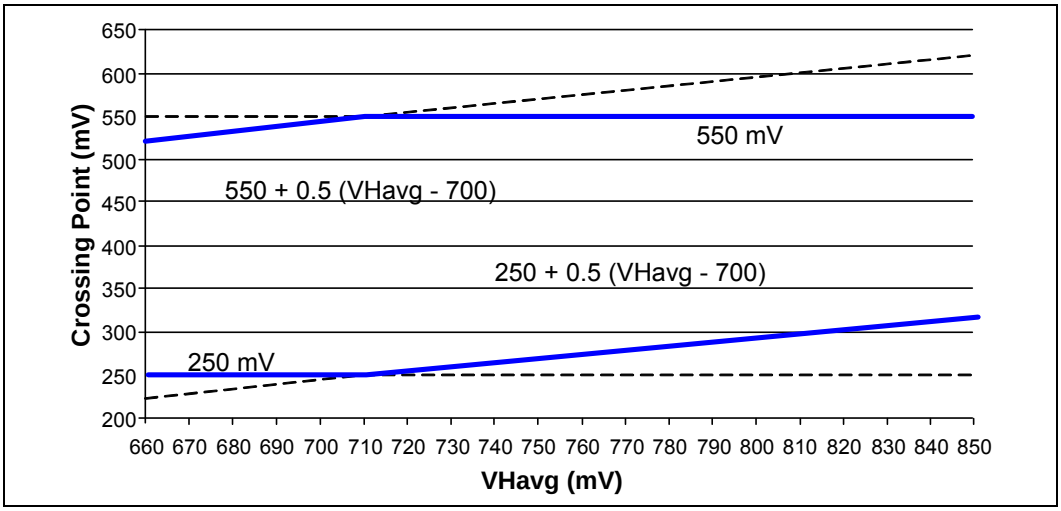


Figure 33-11.Differential Measurement Point for Duty Cycle and Period

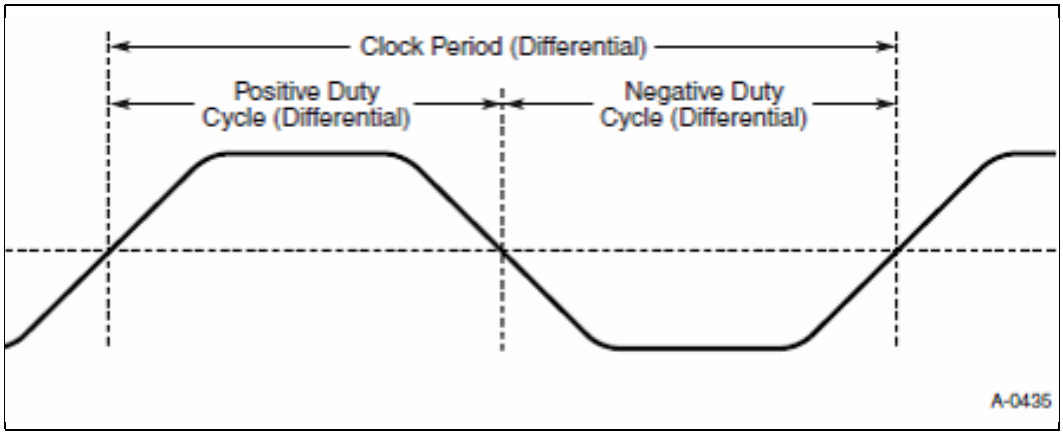


Figure 33-12.Differential Measurement Point for Rise and Fall Time

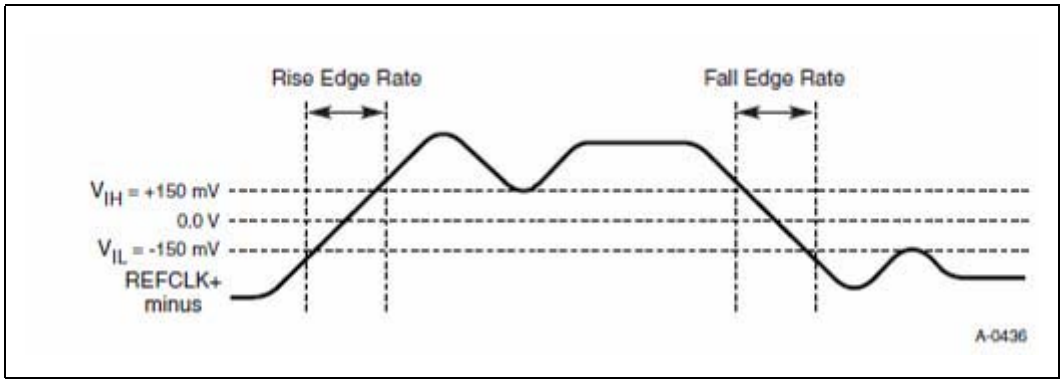
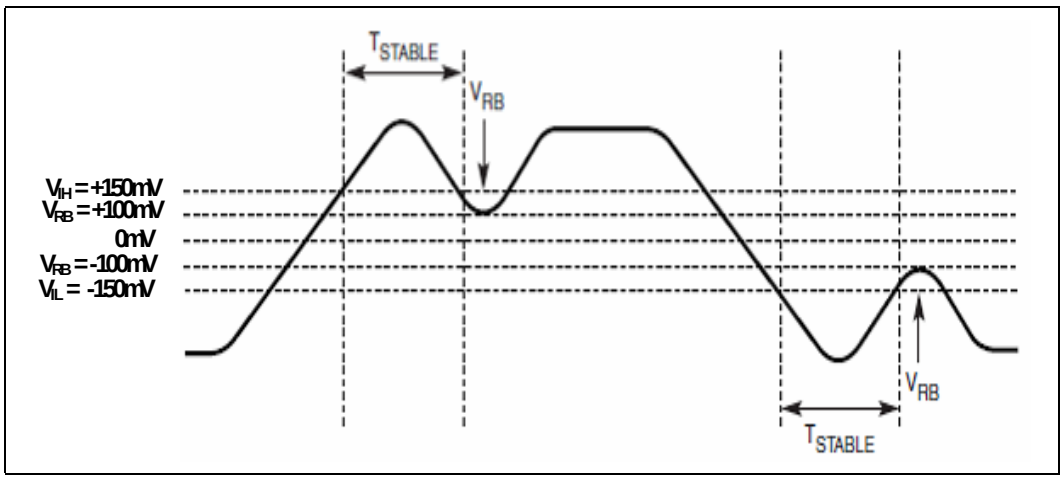


Figure 33-13.Differential Measurement Point for Ringback



33.13 AC Characteristics

Table 33-24. Digital 3.3V I/O Timing Characteristics

Symbol	Parameter	Min	Nom	Max	Unit	Cload
$T_{OUT-RISE}$	Output Rise Time	0.53	0.89	1.27	ns	20pF
$T_{OUT-FALL}$	Output Fall Time	0.56	0.93	1.27	ns	20pF
$T_{OUT-DELAY-RISE}$	Output Delay Rise	1.15	1.96	3.11	ns	20pF
$T_{OUT-DELAY-FALL}$	Output Delay Fall	1.22	2.11	3.33	ns	20pF
$T_{IN-RISE}$	Input Rise Time	0.03		0.1	ns	0.2pF
$T_{IN-FALL}$	Input Fall Time	0.03		0.1	ns	0.2pF
$T_{IN-DELAY-RISE}$	Input Delay Rise	0.3		1.5	ns	0.2pF
$T_{IN-DELAY-FALL}$	Input Delay Fall	0.3		1.5	ns	0.2pF

Notes:
1. Input delay test conditions: Maximum input level = VIN = 2.7V; Input rise/fall time $T_{PAD-RISE}/T_{PAD-FALL}$ (0.2VIN to 0.8VIN) = 1ns (Slew Rate ~ 1.5ns).
2. This table can be used for all 3.3V I/O timing not specified anywhere in this document.

Figure 33-14. Digital I/O Output Timing Diagram

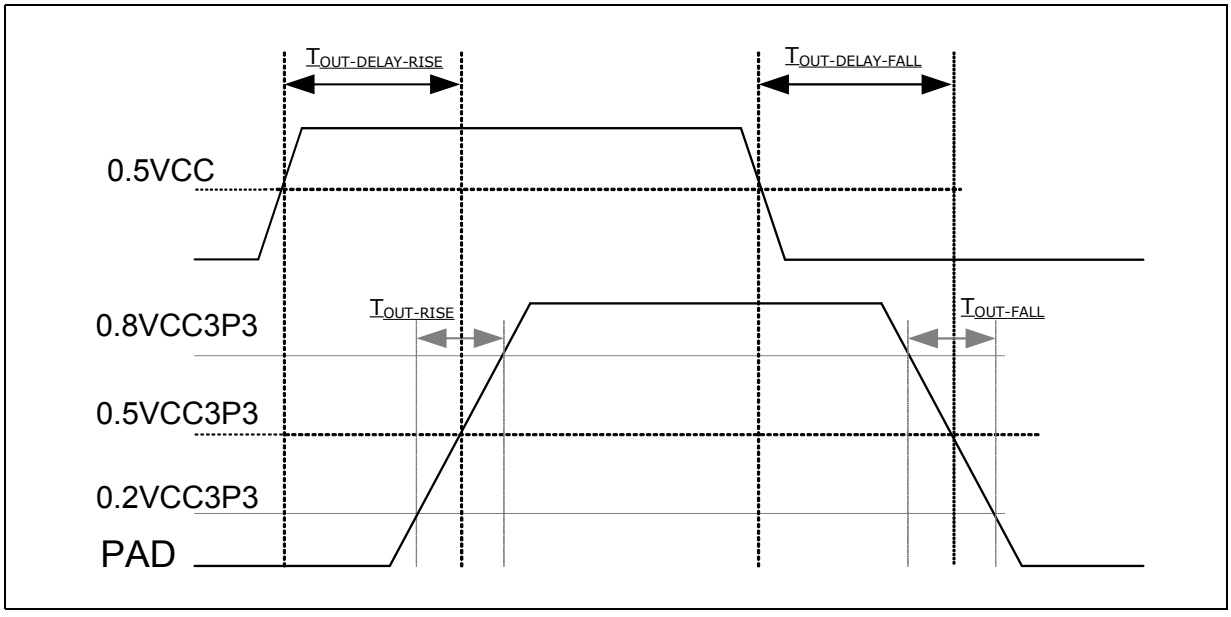


Figure 33-15.Digital I/O Input Timing Diagram

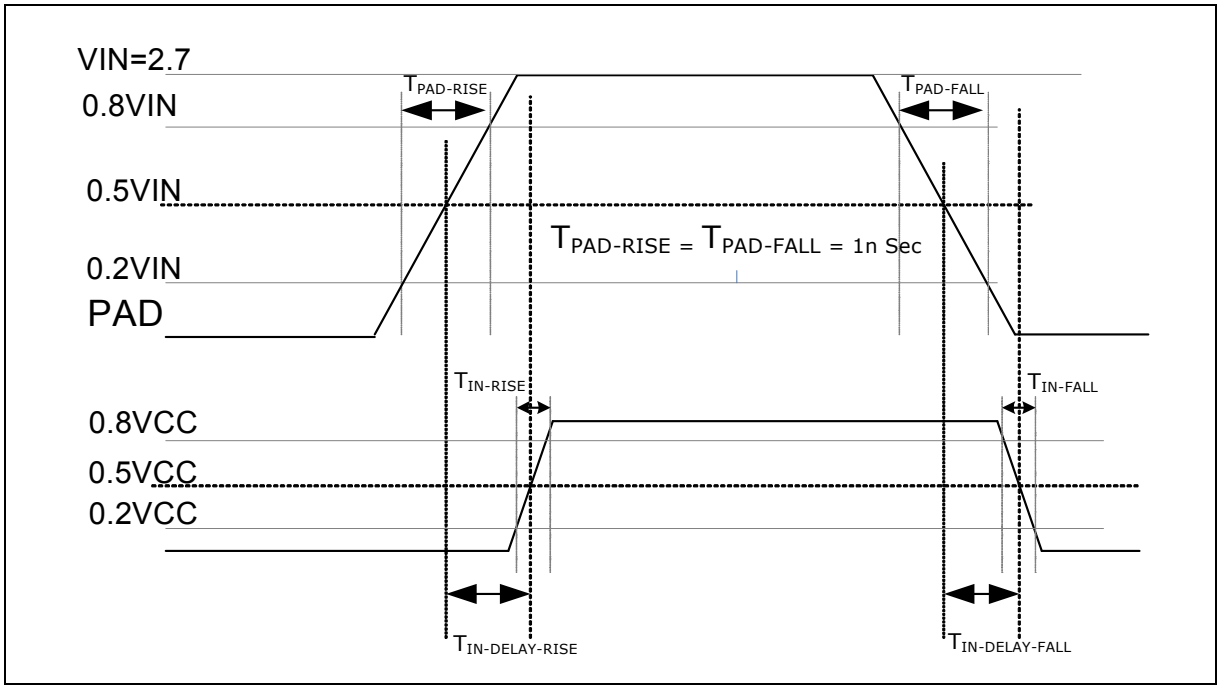


Table 33-25. Clock Timings (Sheet 1 of 3)

Symbol	Parameter	Min	Max	Unit	Notes	Figure
RTC Clock (RTCX1, RTCX2 32.768KHz)						
	Tolerance	-20	20	ppm		
	Duty Cycle	45	55	%		
	High time	10		µs	3	33-22
	Low time	10	—	µs	3	33-22
Cin1, Cin2	Pin Input Capacitance	1	1.2	pF		
UART Clock (UART_CLK 14.7456MHz)						
	Period	67	70	ns		
	High time	20	—	ns		33-22
	Low time	20	—	ns		33-22
	Duty Cycle	45	55	%		
	Rise time	1	4.0	V/ns	3	33-22
	Fall time	1	4.0	V/ns	3	33-22
	Jitter cycle to cycle	—	350	ps	5,6	33-22
LPC Clock (PCICLK 33.00MHz)						

Table 33-25. Clock Timings (Sheet 2 of 3)

Symbol	Parameter	Min	Max	Unit	Notes	Figure
t1	Period	29.566	30.584	ns		
t2	High Time	10.826	17.850	ns		
t3	Low Time	10.426	17.651	ns		33-22
	Duty Cycle	45	55	%		33-22
t4	Rising Edge Rate	1.0	4	V/ns	3	33-22
t5	Falling Edge Rate	1.0	4	V/ns	3	33-22
	Jitter cycle to cycle	—	500	ps	5,6	33-22
Timer (8254) Clock (REF_CLK14 14.31818MHz)						
t6	Period	69.820	69.862	ns		
t7	High Time	29.975	38.467	ns		33-22
t8	Low Time	29.575	38.267	ns		33-22
	Duty Cycle	45	55	%	3	33-22
PPM	Frequency Stability	-500	+500	PPM		
t4	Rising Edge Rate	1.0	4	V/ns	3	33-22
t5	Falling Edge Rate	1.0	4	V/ns	3	33-22
	Jitter cycle to cycle	—	1000	ps	5,6	33-22
UART Clock (UART_CLK 48MHz)						
t9	Period	20.831	20.835	ns	1	
t10	High Time	8.217	11.152	ns		
t11	Low Time	7.817	10.952	ns		33-22
	Duty Cycle	45	55	%		33-22
t4	Rising Edge Rate	1.0	4	V/ns	3	33-22
t5	Falling Edge Rate	1.0	4	V/ns	3	33-22
	Frequency Stability (see the Clock Chip Specification)	-100	+100	PPM		
	Jitter cycle to cycle	—	350	ps	3	33-22
SMBus/SMLink Clock (MST_SMBCLK, EP_SMBCLK, GBE_MST_SMBCLK, SML0CLK)						
f_{smb}	Operating Frequency	10	100	KHz		
t22	High time	4.0	50	µs	2	33-22
t23	Low time	4.7	—	µs		33-22
t24	Rise time	—	1000	ns	3	33-22
t25	Fall time	—	300	ns	3	33-22
GbE Clock (GBE_CLK100[P,N]) from a clock chip						
t36	Period	9.997	10.0533	ns	100MHz	
Duty Cycle	Duty Cycle	45	55			
	Slew rate	1	8	V/ns		
PPM	Frequency Stability (see the Clock Chip Specification)	-50	+50	PPM		

Table 33-25. Clock Timings (Sheet 3 of 3)

Symbol	Parameter	Min	Max	Unit	Notes	Figure
	Aging	-5	+5	PPM per Year		
	Voltage Swing (HCSL)	0.55	0.85	Vpp		
V_{ISE}	Voltage Input Single Ended	-0.1	1.15	V		
	Jitter Cycle to cycle		80	ps	5,6	
USB Clock (USB_CLK96[P,N]) from a clock chip						
t36	Period	10.066	10.768	ns		
	Slew rate	1	8	V/ns	4	
	Frequency Stability (see the Clock Chip Specification)	-100	+100	PPM		
	Voltage Swing (HCSL)	0.55	0.85	Vpp		
V_{ISE}	Voltage Input Single Ended	-0.1	1.15	V		
	Jitter cycle to cycle	—	250	ps	5,6	
Suspend Clock (SUS_CLK)						
f_{sus_clk}	Operating Frequency	32.768		kHz	2	
t39	High Time	10	—	µs	2	
t39a	Low Time	10	—	µs	2	
SPI Clock (SPI_CLK)						
Slew_Rise	Output Rise Slew Rate (0.2VCC3P3 - 0.6VCC3P3)	1	4	V/ns	1	
Slew_Fall	Output Fall Slew Rate (0.6VCC3P3 - 0.2VCC3P3)	1	4	V/ns	1	

1. The UART_CLK @48MHz expects a 40/60% duty cycle.
2. SUS_CLK duty cycle can range from 30% minimum to 70% maximum.
3. Rise and Fall time is measured 20% and 80% of signal edge.
4. See *CK420 Clock Synthesizer Specification* for measurement procedure.
5. Jitter is specified as cycle to cycle measured in pico seconds. Period min and max includes cycle to cycle jitter.
6. On all jitter measurements care should be taken to set the zero crossing voltage (for rising edge) of the clock to be the point where the edge rate is the fastest. Using a Math function = Average (Derivative (Ch1)) and set the averages to 64, place the cursors where the slope is the highest on the rising edge - usually this lower half of the rising edge. The reason this is defined is for users trying to measure in a system it is impossible to get the probe exactly at the end of the Transmission line with large Flip Chip components, this results in a reflection induced ledge in the middle of the rising edge and will significantly increase measured jitter.
7. Phase jitter requirement: The designated Gen2 outputs will meet the reference clock jitter requirements from the PCI Express Gen2 Base Spec. The test is to be performed on a component test board under quiet conditions with all clock outputs on. Jitter analysis is performed using a standardized tool provided by the PCI SIG. Measurement methodology is defined in Intel document “PCI Express Reference Clock Jitter Measurements”.
8. Testing condition: 1K Ω pull up to VCC3P3, 1K Ω pull down and 10pF pull down and 1/2 inch trace.

Table 33-26. Universal Serial Bus (USB) Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
High-speed Source (Note 7)						
t100	USB[5:0]Dp, USB[5:0]Dn Driver Rise Time	0.8	1.2	ns	1, C_L = 10 pF	
t101	USB[5:0]Dp, USB[5:0]Dn Driver Fall Time	0.8	1.2	ns	1, C_L = 10 pF	
Full-speed Source (Note 8)						
t102	USB[5:0]Dp, USB[5:0]Dn Driver Rise Time	4	20	ns	1, C_L = 50 pF	
t103	USB[5:0]Dp, USB[5:0]Dn Driver Fall Time	4	20	ns	1, C_L = 50 pF	
t104	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	 −3.5 −4	 3.5 4	 ns ns	2, 3	33-29
t105	Source SE0 interval of EOP	160	175	ns	4	33-30
t108	EOP Width: Must accept as EOP	82	—	ns	2	33-30
Low-speed Source (Note 9)						
t110	USB[5:0]Dp, USB[5:0]Dn Driver Rise Time	75	300	ns	1, 6 C_L = 50 pF C_L = 350 pF	
t111	USB[5:0]Dp, USB[5:0]Dn Driver Fall Time	75	300	ns	1,6 C_L = 50 pF C_L = 350 pF	
t112	Source Differential Driver Jitter To Next Transition For Paired Transitions	 −25 −14	 25 14	 ns ns	2, 3	33-29
t114	Source Jitter for Differential Transition to SE0 Transition	−40	100	ns	5	
t116	EOP Width: Must accept as EOP	670	—	ns	2	33-30

Notes:
1. Driver output resistance under steady state drive is spec'd at 28 Ω at minimum and 43 Ω at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. High-speed Data Rate has minimum of 479.760 Mb/s and maximum of 480.240 Mb/s.
8. Full-speed Data Rate has minimum of 11.97 Mb/s and maximum of 12.03 Mb/s.
9. Low-speed Data Rate has a minimum of 1.48 Mb/s and a maximum of 1.52 Mb/s.

Table 33-27. SMBus Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
t131	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4.0	—	µs		33-31
t132	Repeated Start Condition Setup Time	4.7	—	µs		33-31
t133	Stop Condition Setup Time	4.0	—	µs		33-31
t134	Data Hold Time	0	—	ns	1	33-31
t135	Data Setup Time	250	—	ns		33-31

1. t134 has a minimum timing for I^2C of 0 ns, while the minimum timing for SMBus is 300 ns.

Table 33-28. LPC Timing

Sym	Parameter	Min	Max	Units	Notes	Fig
t150	LAD[3:0] Valid Delay from PCICLK Rising	2	13	ns		33-23
t151	LAD[3:0] Output Enable Delay from PCICLK Rising	2		ns		33-27
t152	LAD[3:0] Float Delay from PCICLK Rising	—	28	ns		33-25
t153	LAD[3:0] Setup Time to PCICLK Rising	7	—	ns		33-24
t154	LAD[3:0] Hold Time from PCICLK Rising	0	—	ns		33-24
t155	LDRQ[1:0]# Setup Time to PCICLK Rising	12	—	ns		33-24
t156	LDRQ[1:0]# Hold Time from PCICLK Rising	0	—	ns		33-24
t157	LFRAME# Valid Delay from PCICLK Rising	2	13	ns		33-23

Table 33-29. SPI Timings (20 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
t180a	Serial Clock Frequency - 20M Hz Operation	17.2	18.4	MHz	1	
t182a	SPI Clock Duty cycle at the host	43	57	%		33-33
t183a	Tco of SPI_MOSI with respect to serial clock falling edge at the host	-5	13	ns		33-33
t184a	Setup of SPI_MISO with respect to serial clock falling edge at the host	16	—	ns		33-33
t185a	Hold of SPI_MISO with respect to serial clock falling edge at the host	0	—	ns		33-33
t186a	Setup of SPI_CS[1:0]# assertion with respect to serial clock rising at the host	30	—	ns		33-33
t187a	Hold of SPI_CS[1:0]# deassertion with respect to serial clock falling at the host	30	—	ns		33-33
t188a	SPI_CLK high time	26.37	—	ns	2	33-33
t189a	SPI_CLK low time	26.82	—	ns	2	33-33

Note:
1. The typical clock frequency driven by the PCH is 17.86 MHz.
2. Measurement point for low time and high time is taken at 0.5 (VCC3P3)

Table 33-30. SPI Timings (33 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
t180b	Serial Clock Frequency - 33 MHz Operation	30.3	32.19	MHz	1	
t182b	SPI Clock Duty cycle at the host	50	50	%		33-33
t183b	Tco of SPI_MOSI with respect to serial clock falling edge at the host	-5	5	ns		33-33
t184b	Setup of SPI_MISO with respect to serial clock falling edge at the host	8	—	ns		33-33
t185b	Hold of SPI_MISO with respect to serial clock falling edge at the host	0	—	ns		33-33

Table 33-30. SPI Timings (33 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
t186b	Setup of SPI_CS[1:0]# assertion with respect to serial clock rising at the host	30	—	ns		33-33
t187b	Hold of SPI_CS[1:0]# deassertion with respect to serial clock falling at the host	30	—	ns		33-33
t188b	SPI_CLK high time	14.88	—	ns	2	33-33
t189b	SPI_CLK low time	15.18	—	ns	2	33-33

Note:
1. The typical clock frequency driven by the PCH is 31.25 MHz.
2. Measurement point for low time and high time is taken at 0.5 (VCC3P3)

Table 33-31. SPI Timings (50 MHz)

Sym	Parameter	Min	Max	Units	Notes	Fig
t180c	Serial Clock Frequency - 50MHz Operation	46.99	53.40	MHz	1	33-33
t182c	SPI Clock Duty cycle at the host	40	60	%		33-33
t183c	Tco of SPI_MOSI with respect to serial clock falling edge at the host	-3	3	ns		33-33
t184c	Setup of SPI_MISO with respect to serial clock falling edge at the host	8	-	ns		33-33
t185c	Hold of SPI_MISO with respect to serial clock falling edge at the host	0	-	ns		33-33
t186c	Setup of SPI_CS[1:0]# assertion with respect to serial clock rising edge at the host	30	-	ns		33-33
t187c	Hold of SPI_CS[1:0]# assertion with respect to serial clock rising edge at the host	30	-	ns		33-33
t188c	SPI_CLK High time	7.1	-	ns	2, 3, 4	33-33
t189c	SPI_CLK Low time	11.17	-	ns	2, 3	33-33

1. Typical clock frequency driven by the PCH is 50 MHz. This frequency is not available for ES1 samples.
2. When using 50 MHz mode ensure target flash component can meet t188c and t189c specifications.
3. Measurement point for low time and high time is taken at 0.5(VCC3P3)
4. When operating at 50MHz, the PCH uses a divide down from 125MHz clock. Due to the 40% duty cycle, the PCH SPI Flash Controller cannot meet the minimum high timing requirements of a 50MHz SPI Flash component and a 66MHz rate or faster SPI Flash component must be used.

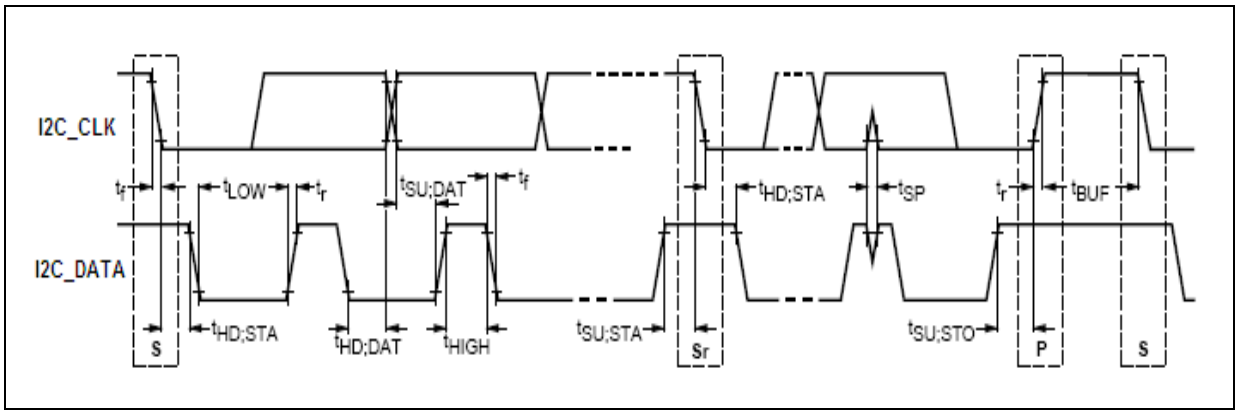
33.13.1 I²C/SFP AC Specification

Table 33-32 indicates the timing of the SFPn_I2C_CLK and SFPn_I2C_DATA pins when operating in I²C mode.

Table 33-32. I²C Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units
$T_{HD:DAT}$	Data hold time	0.3			µs
$T_{SU:DAT}$	Data setup time	0.25			µs

Figure 33-16. I²C I/F Timing Diagram



33.13.2 MDIO DC Specification

Table 33-33, "DC Input Characteristics: MDIO Mode of Operation" and Table 33-34, "DC Output Characteristics: MDIO Mode of Operation" indicate the DC and timing of the SFPn_I2C_CLK and SFPn_I2C_DATA pins when operating in MDIO mode.

Table 33-33. DC Input Characteristics: MDIO Mode of Operation

Symbol	Parameter	Conditions	Min	Nominal	Max	Units	Notes
V_{IH}	Input high voltage	$V_{IH} > V_{IH\_Min}$ V_{CC} = Min	2.0	-	3.6	V	-
V_{IL}	Input low voltage	$V_{IH} > V_{IL\_Max}$ V_{CC} = Min	-	-	0.8	V	-
I_{IH}	Input high current	V_{CC} = Max V_{IN} = 2.5 V	-	-	15	uA	-
I_{IL}	Input low current	V_{CC} = Max V_{IN} = 0.4 V	-15	-	-	uA	-
I_{leak}	Input Leakage Current	0<VIN< VCCPEP3P3AUX	-10	-	10	µA	-
C_{IN}	Input pin cap		-	-	8	pF	1

Notes:
1. Guaranteed by design. These values are typical values seen for this process, but not measured during production testing.

Table 33-34. DC Output Characteristics: MDIO Mode of Operation

Symbol	Parameter	Conditions	Min	Typical	Max	Units	Notes
V_{OH}	Output high voltage	V_{CC} = 3.3V I_{OH} = -1mA	2.4	-	VCCPEP3P3AUX	V	1
V_{OH}	Output high voltage	V_{CC} = 3.3V I_{OH} = -4mA	2.4	-	VCCPEP3P3AUX	V	1
V_{OL}	Output low voltage	V_{CC} = Min I_{OL} = 1mA	-	-	0.4	V	

Notes:
1. The MDIO buffer is powered from the 3.3V Auxiliary power rail.

33.13.3 MDIO AC Specification

The PCH is designed to support the MDIO specifications defined in IEEE 802.3, clause 22.

Table 33-35 contains specifications applicable over a recommended operating range at VCC3P3 = 3.3V and Cload = 16pF (unless otherwise noted). For MDIO I/F timing specification, see Table 33-35, Figure 33-17, and Figure 33-18.

Table 33-35. MDIO I/F Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units	Note
t_{MCLK}	MDC clock period (2 MHz)			500	nS	1, 2
t_{MHI}	MDIO input hold time after the rising edge of MDC clock	0			nS	1, 2
t_{MHO}	MDIO output hold time after the rising edge of MDC clock	10			nS	1, 2
t_{MSU}	MDIO setup time	10			nS	1, 2
t_{MPR}	MDIO propagation Delay	10		TMCLK/2 +10	nS	1, 2

Notes:
1. This table applies to MDIO0, MDC0, MDIO1, MDC1, MDIO2, MDC2, MDIO3, and MDC3.
2. Timing measured relative to MDC reference voltage of 2.0V (Vih).

Figure 33-17. MDIO Input AC Timing Diagram

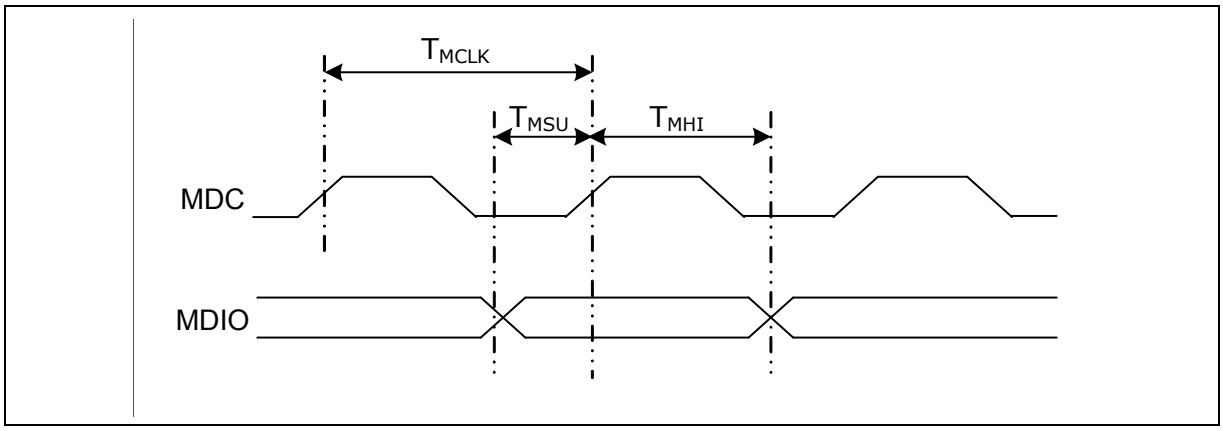
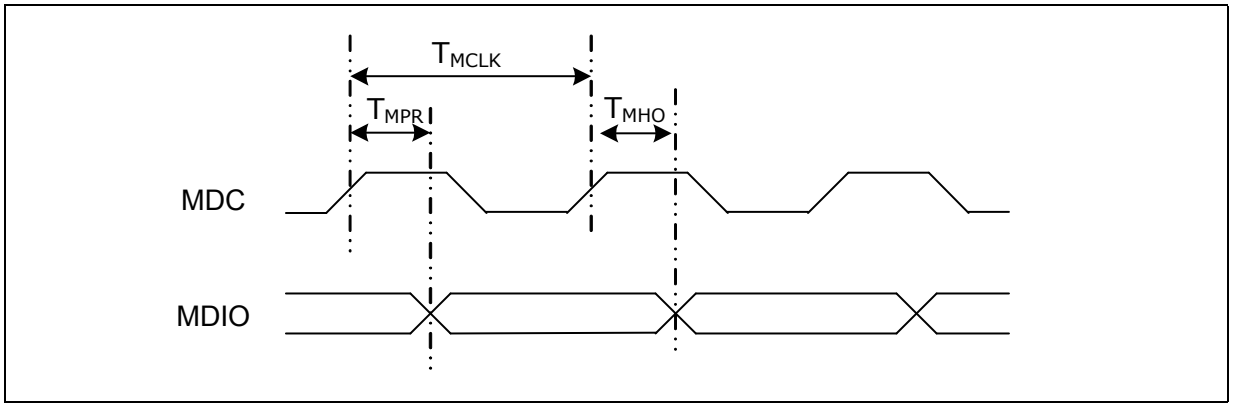


Figure 33-18.MDIO Output AC Timing Diagram



33.13.4 EEPROM AC Specification

The PCH is designed to support a standard serial EEPROM. The following timing specifications are applicable over a recommended operating range at VCC3P3 = 3.3V and Cload = 16pF unless otherwise noted. For EEPROM I/F timing specification, see Table 33-36 and Figure 33-19.

Table 33-36. EEPROM I/F Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units	Note
t_{SCK}	GBE_EE_SK/EP_EE_SK clock frequency	0	2	2.1	MHz	1
t_{RI}	Input rise time			2	µs	
t_{FI}	Input fall time			2	µs	
t_{WH}	GBE_EE_SK/EP_EE_SK high time	200	250		ns	2
t_{WL}	GBE_EE_SK/EP_EE_SK low time	200	250		ns	
t_{CS}	GBE_EE_CS#/EP_EE_CS# high time	250			ns	
t_{CSS}	GBE_EE_CS#/EP_EE_CS# setup time	250			ns	
t_{CSH}	GBE_EE_CS#/EP_EE_CS# hold time	250			ns	
t_{SU}	Data-in setup time	50			ns	
t_{H}	Data-in hold time	50			ns	
t_{V}	Output valid from Clock Low	0		200	ns	
t_{HO}	Output hold time	0			ns	
t_{DIS}	Output disable time			250	ns	

Notes:
1. Clock is 2MHz
2. 50% duty cycle.

Figure 33-19.EEPROM Timing Diagram

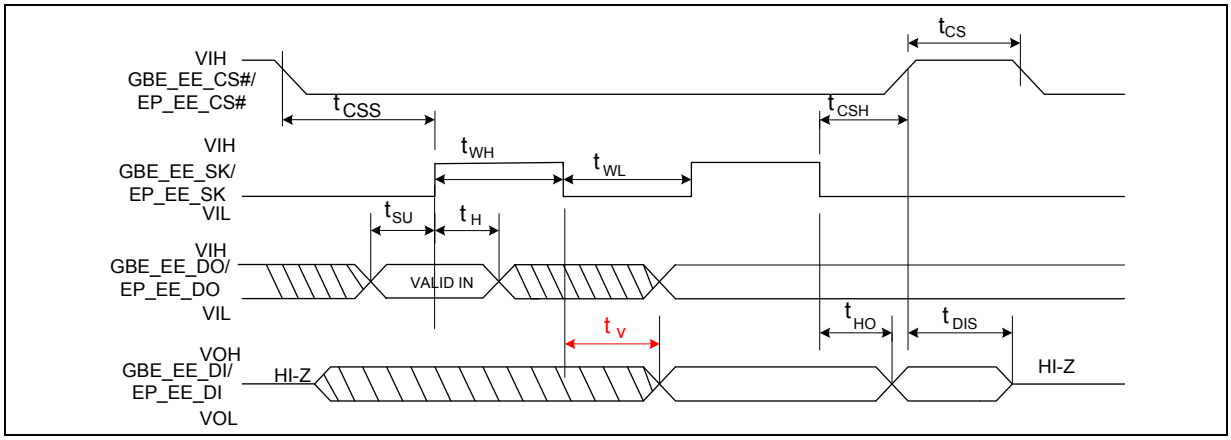


Table 33-37. PECI Timings

Sym	Parameter	Min	Max	Units	Notes
t_{BIT}	Bit time (overall time evident on PECI) Bit time driven by an originator	0.495 0.495	500 250	µs µs	1
t_{BIT},jitter	Bit time jitter between adjacent bits in an PECI message header or data bytes after timing has been negotiated	—	—	%	
t_{BIT},drift	Change in bit time across a PECI address or PECI message bits as driven by the originator. This limit only applies across t_{BIT-A} bit drift and t_{BIT-M} drift.	—	—	%	4
t_{H1}	High level time for logic '1'	0.6	0.8	x t_{BIT}	2
t_{H0}	High level time for logic '0'	0.2	0.4	x t_{BIT}	
t_{PECIR}	Rise time (measured from VOL to VIH,min, Vtt(nom) -5%)	—	30 + 5	ns/node	3
t_{PECIF}	Fall time (measured from V_{OH} to V_{IL},max, Vtt(nom) +5%)	—	30	ns/node	3

Notes:
1. The originator must drive a more restrictive time to allow for quantized sampling errors by a client yet still attain the minimum time less than 500 µs. t_{BIT} limits apply equally to t_{BIT-A} and t_{BIT-M}. PCH is targeted on 2 MHz which is 500 ns bit time.
2. The minimum and maximum bit times are relative to t_{BIT} defined in the Timing Negotiation pulse.
3. Extended trace lengths may appear as additional nodes.
4. t_{BIT-A} is the negotiated address bit time and t_{BIT}–M is the negotiated message bit time.

33.13.5 UART AC Specification

This section describes the Universal Asynchronous Receiver/Transmitter (UART) serial port used for the two UARTs that are integrated into the Serial I/O unit and Watchdog Timer (SIW). The UART can be controlled via programmed I/O. The basic programming model is the same for both UARTs, with the only difference being the Logical Device Number assigned to each. The serial port consists of a UART that supports all the functions of a standard 16550 UART, including hardware flow control interface.

Table 33-38. UART Received Timing

Symbol	Parameter	Min	Typ	Max	Units	Note
t_{SetUp}	Set Up time from the rising edge of UART_CLK	7			nS	1
t_{Hold}	Hold time from the rising edge of UART_CLK	0			nS	1

Notes:
1. Applies to SIU[1:0]_RX, SIU[1:0]_CTS#, SIU[1:0]]_DSR#, SIU[1:0]_DCD#, SIU[1:0]_SI#

Table 33-39. UART Transmit Timing

Symbol	Parameter	Min	Typ	Max	Units	Note
$t_{Valid1-Delay1}$	Valid Output Delay 1 from the rising edge of UART_CLK	2		11	nS	1
$t_{Valid-Delay2}$	Valid Output Delay 2 from the rising edge of UART_CLK	2			nS	2

Notes:
1. Applies to SIU[1:0]_TX
2. Applies to SIU[1:0]DTR#, SIU[1:0]_RST#

33.13.6 JTAG AC Specification

The PCH is designed to support the IEEE 1149.1 standard. For JTAG I/F timing specification, see Table 33-40 and Figure 33-20.

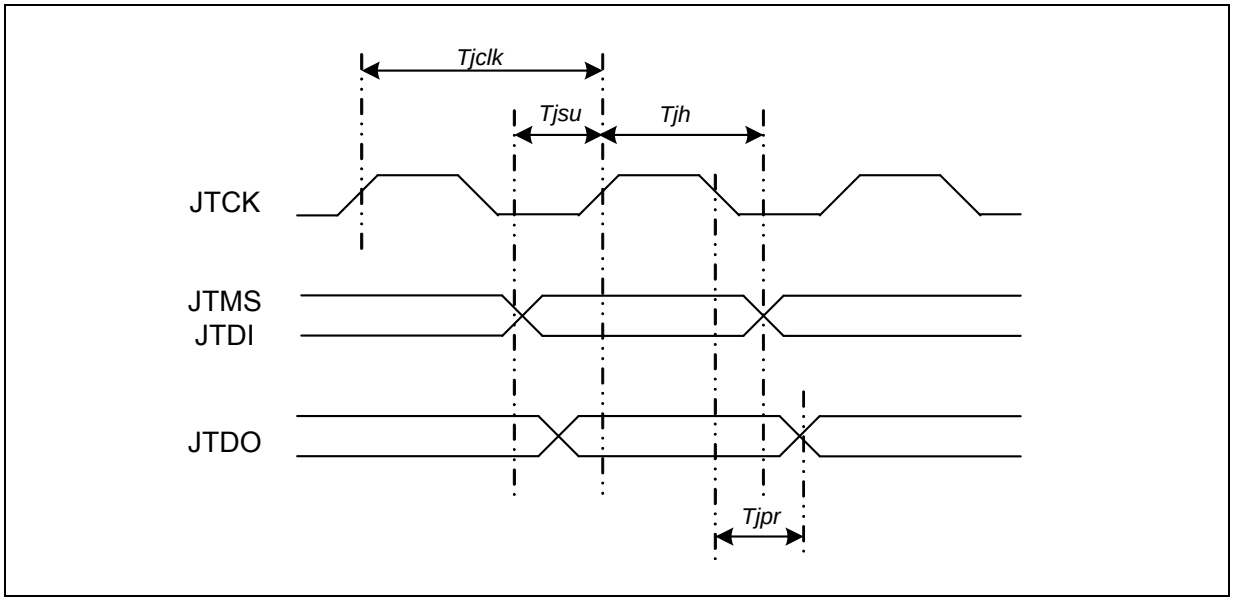
Table 33-40. JTAG I/F Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units	Note
t_{JCLK}	JTCK/EP_JTCK clock frequency			10	MHz	1
t_{JH}	JTMS/EP_TMS and JTDI/EP_TDI hold time	10			nS	
t_{JSU}	JTMS/EP_TMS and JTDI/EP_TDI setup time	10			nS	
t_{JPR}	JTDO/EP_TDO propagation Delay			15	nS	

1. Timing measured relative to JTCK reference voltage of VCC3P3/2.

Note: Table 33-40 applies to JTCK, JTMS, JTDI and JTDO.

Figure 33-20.JTAG AC Timing Diagram



33.14 AC Timing Diagrams

Figure 33-21.Clock Cycle Time

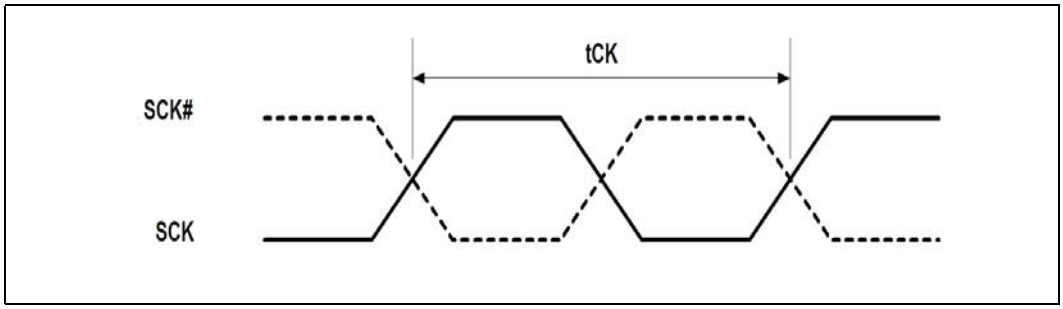


Figure 33-22.Clock Timing

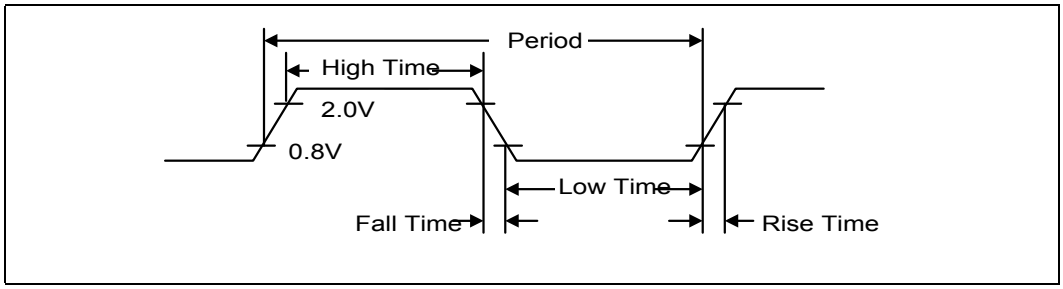


Figure 33-23.Valid Delay from Rising Clock Edge

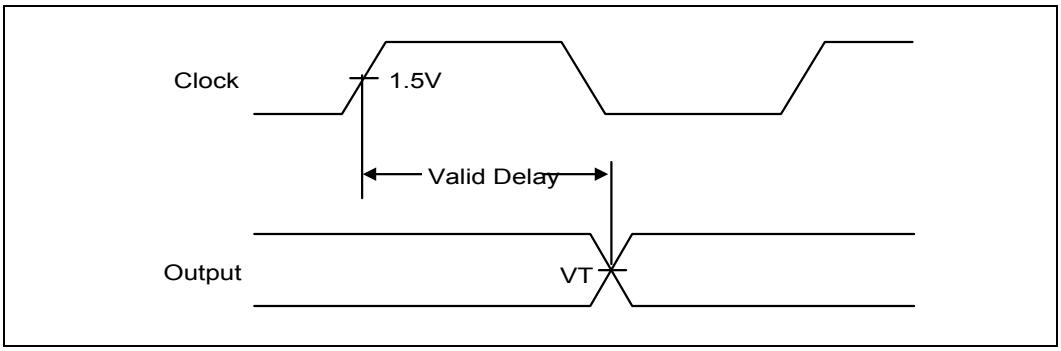


Figure 33-24.Setup and Hold Times

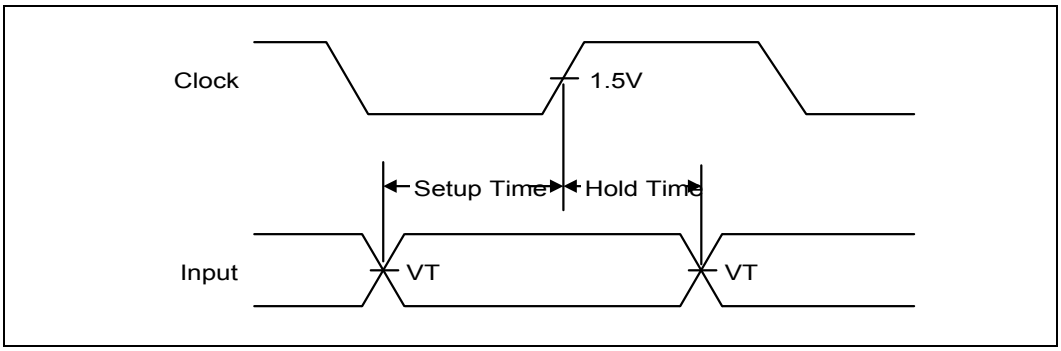


Figure 33-25.Float Delay

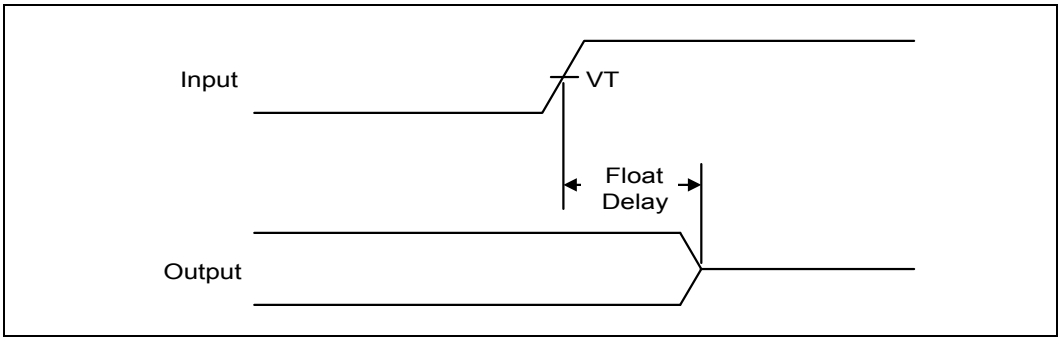


Figure 33-26.Pulse Width

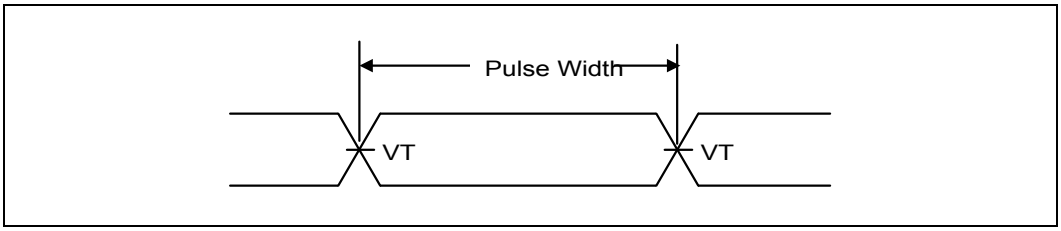


Figure 33-27.Output Enable Delay

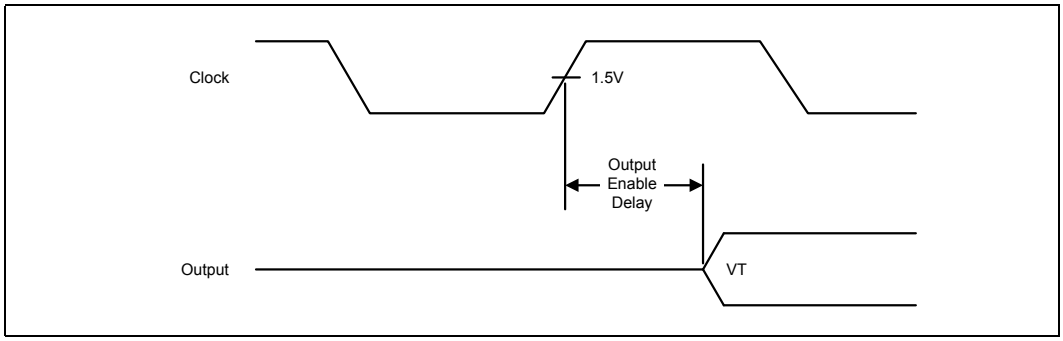


Figure 33-28.USB Rise and Fall Times

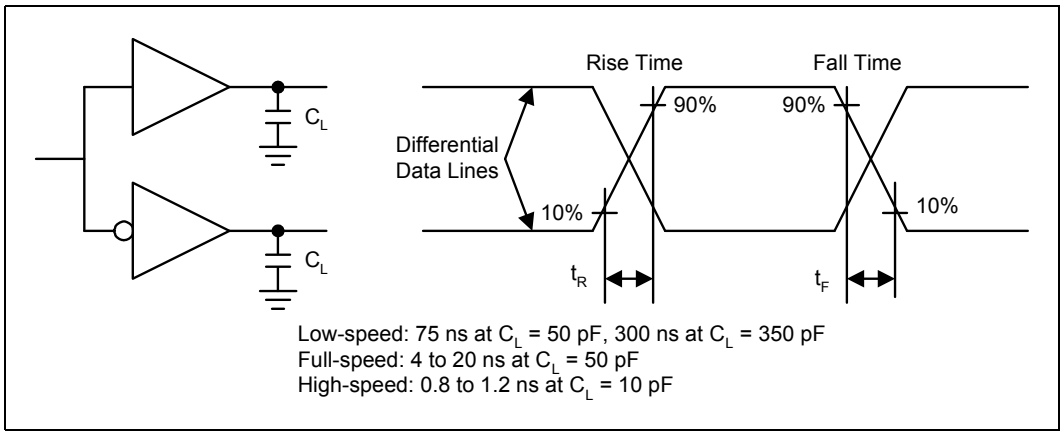


Figure 33-29.USB Jitter

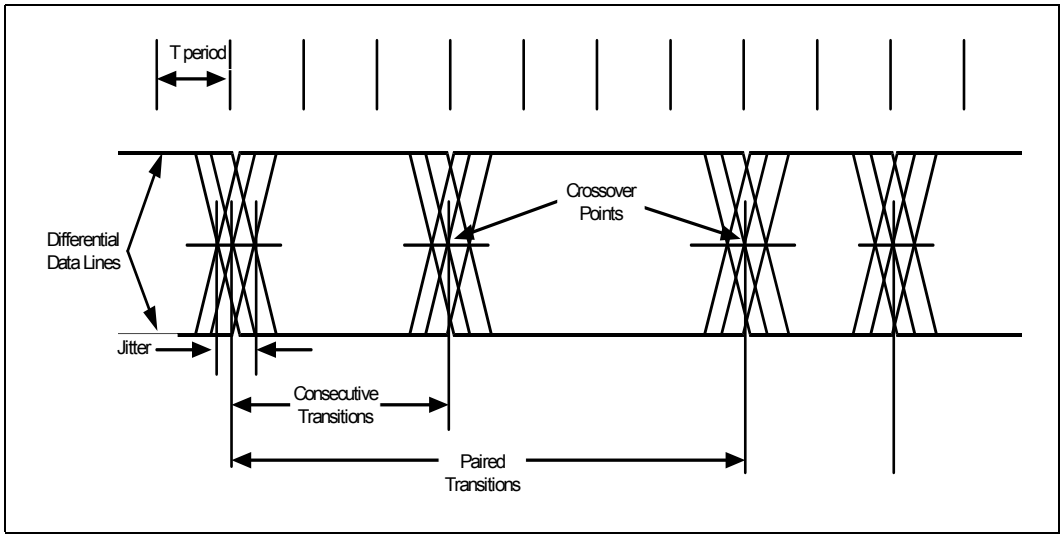


Figure 33-30.USB EOP Width

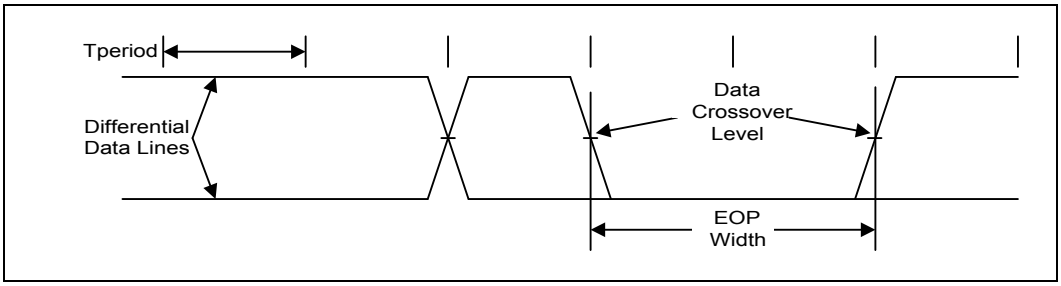


Figure 33-31.SMBus Transaction

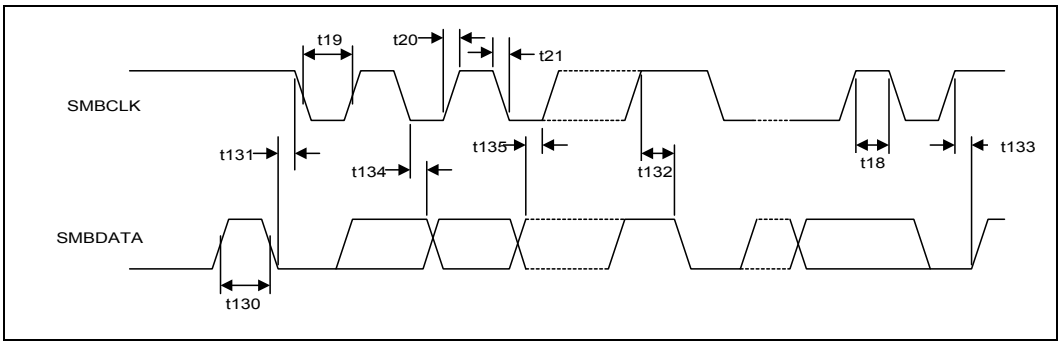


Figure 33-32.SMBus Timeout

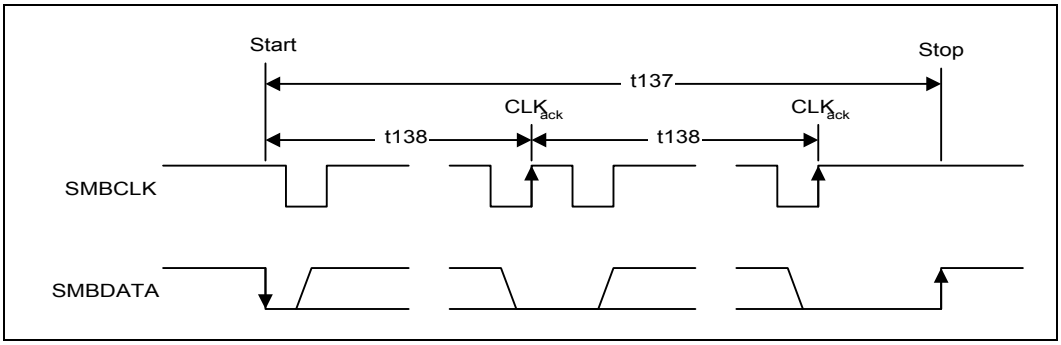
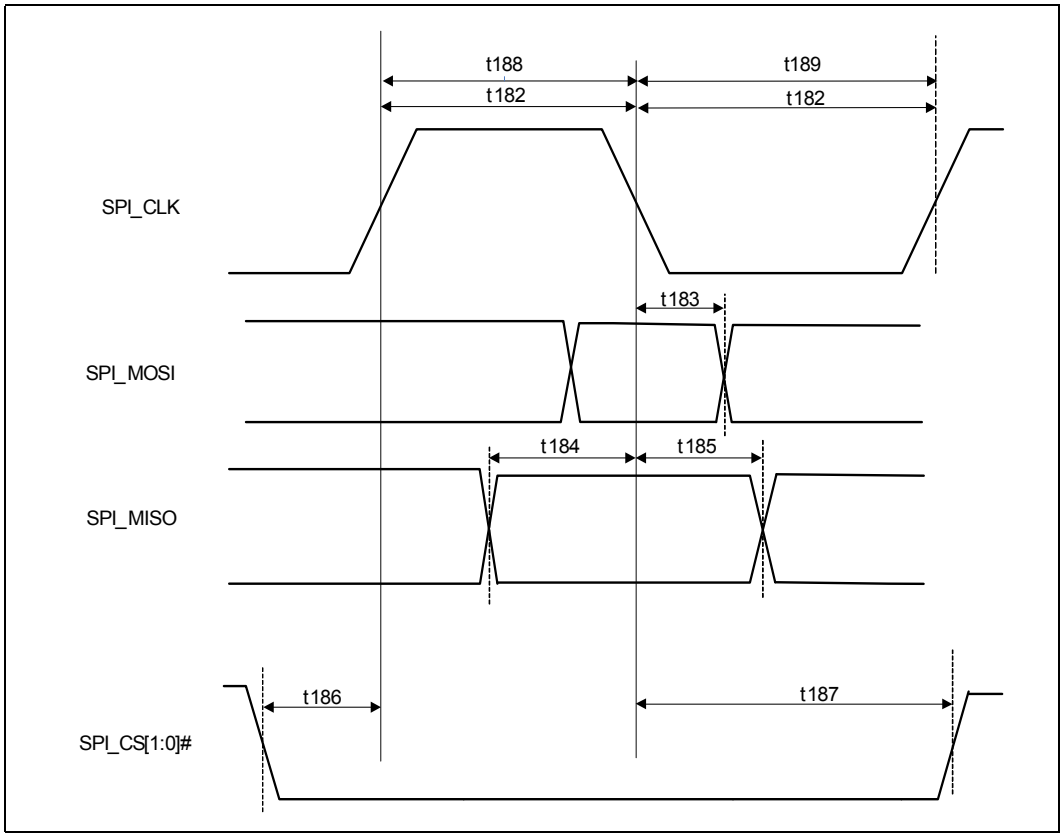
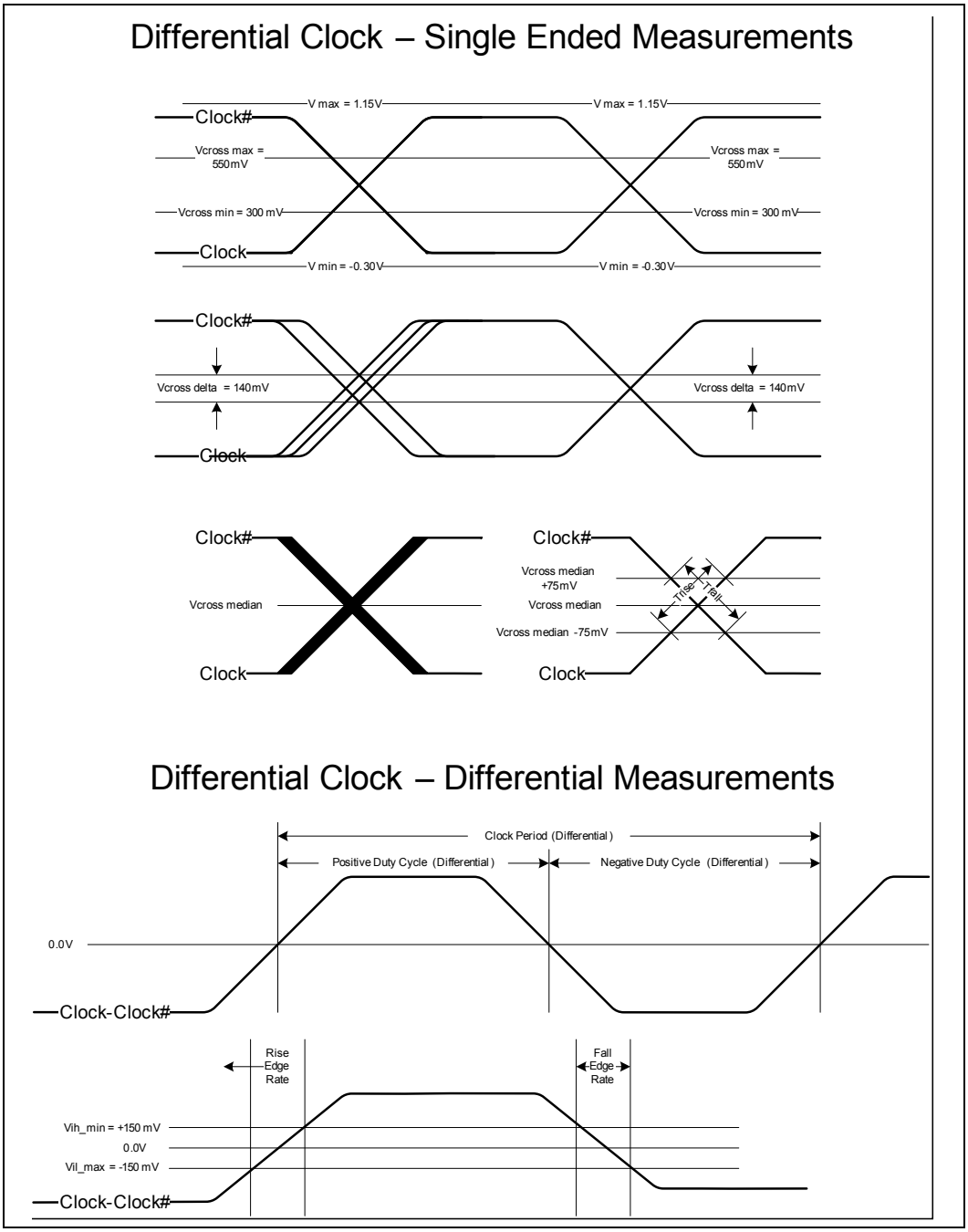


Figure 33-33.SPI Timings



Note: t184 and t185 are referencing clock edge where the PCH will sample the SPI_MISO pin. The slave transmitted this bit on the previous falling clock edge which is not shown.

Figure 33-34.Measurement Points for Differential Waveforms



§ §

34.0 Thermal Specifications and Design Considerations

For thermal specifications and design guidelines, see the *Intel® Communications Chipset 89xx, 8903, 8910, and 8920 Thermal and Mechanical Design Guide* and the *Intel® Communications Chipset 8925, 8950, and 8955 Thermal and Mechanical Design Guide.*

§ §

35.0 Packaging Information

35.1 Package Introduction

The product line is offered with the following specifications:

- Flip-Chip Ball Grid Array (FCBGA)
- Package Size: 27mm x 27mm
- Ball Count: 942
- Ball Pitch: 0.7mm

35.2 Ball Map Information

This section contains the following ball map information:

- Table 35-1, ”Alphabetical Ball Listing”
- Table 35-2, ”Alphabetical Signal Listing”
- Table 35-3, ”Ball Map Color Code”
- Table 35-4, ”Bottom View Left”
- Table 35-5, ”Bottom View Right”
- Table 35-6, ”Top View Left”
- Table 35-7, ”Top View Right”

35.2.1 Ball Map Pin Lists

Table 35-1. Alphabetical Ball Listing

Ball	Signal
A4	VSS
A6	RSVD8
A9	VSS
A12	EP_JTCK
A16	VSS
A19	VSS
A22	RSVD27
A26	VSS
A29	VSS
A32	VSS
A36	VSS
A39	VSS
A42	VSS
A46	VSS
A49	VSS
A52	VSS
A54	VSS
B2	RSVD2
B5	RSVD1
B7	GBE_AUX_PWR_AVAIL
B11	RSVD81
B13	EP_SMBCLK
B15	RSVD5
B17	RSVD11
B21	RSVD21
B23	RSVD25
B25	RSVD29
B27	PCIE_EP_RXp15
B31	PCIE_EP_RXp13
B33	PCIE_EP_RXp12
B35	PCIE_EP_RXn11
B37	PCIE_EP_RXn10
B41	PCIE_EP_RXp7
B43	PCIE_EP_RXp6
B45	PCIE_EP_RXn5
B47	PCIE_EP_RXn4
B51	PCIE_EP_RXn3
B53	VSS
B56	VSS

Ball	Signal
C4	RSVD6
C6	SRDS1_SD
C9	RSVD79
C12	EP_SMBALRT#
C16	RSVD13
C19	RSVD28
C22	VSS
C26	VSS
C29	PCIE_EP_RXp14
C32	VSS
C36	PCIE_EP_RXp10
C39	PCIE_EP_RXn9
C42	PCIE_EP_RXn7
C46	PCIE_EP_RXp4
C49	VSS
C52	VSS
C54	RSVD52
D1	RSVD4
D3	VSS
D5	GBE_EE_DI
D7	VSS
D13	EP_JTRST#
D15	RSVD22
D23	RSVD23
D25	RSVD31
D33	PCIE_EP_RXn12
D35	PCIE_EP_RXp11
D43	PCIE_EP_RXn6
D45	PCIE_EP_RXp5
D47	VSS
D51	PCIE_EP_RXp3
D53	TS0_IREF_N
D55	RSVD51
D57	RSVD55
E2	RSVD0
E4	GBE2_LED
E6	VSS
E9	RSVD82
E11	RSVD80

Ball	Signal
E17	RSVD26
E19	VSS
E21	RSVD34
E27	PCIE_EP_RXn15
E29	PCIE_EP_RXn14
E31	PCIE_EP_RXn13
E37	PCIE_EP_RXp9
E39	PCIE_EP_RXp8
E41	PCIE_EP_RXn8
E49	VSS
E52	VSS
E54	VSS
E56	RSVD54
F1	RSVD3
F3	SRDS2_SD
F5	RSVD85
F8	GBE_EE_DO
F10	PCIE_EP_RST#
F12	VSS
F13	RSVD16
F15	VSS
F16	VSS
F18	RSVD9
F20	RSVD30
F22	RSVD18
F23	RSVD19
F25	VSS
F26	VSS
F28	VSS
F30	VSS
F32	VSS
F33	VSS
F35	VSS
F36	VSS
F38	VSS
F40	VSS
F42	VSS
F43	VSS
F45	VSS

Ball	Signal
F46	VSS
F48	VSS
F50	VSS
F53	VSS
F55	VSS
F57	VSS
G2	VSS
G4	GBE2_SWDP1
G54	PCIE_EP_RXn2
G56	PCIE_EP_RXp2
H6	RSVD84
H8	RSVD86
H10	GBE3_SWDP1
H12	GBE_WAKE#
H15	VSS
H17	RSVD7
H20	VSS
H22	RSVD17
H25	RSVD35
H27	RSVD32
H31	PCIE_EP_TXn14
H33	VSS
H36	PCIE_EP_TXn10
H38	PCIE_EP_TXn9
H41	PCIE_EP_TXp7
H43	VSS
H46	PCIE_EP_TXp5
H48	PCIE_EP_TXp4
H50	PCIE_EP_TXn4
H52	VSS
J1	RSVD83
J3	GBE1_SWDP1
J5	VSS
J10	GBE_EE_SK
J12	VSS
J15	RSVD89
J17	EP_JTMS
J20	RSVD14
J22	CRU_CLK100N
J25	CRU_CLK100P
J27	VSS

Ball	Signal
J31	PCIE_EP_TXp14
J33	PCIE_EP_TXp12
J36	PCIE_EP_TXp10
J38	PCIE_EP_TXp9
J41	PCIE_EP_TXn7
J43	PCIE_EP_TXp6
J46	PCIE_EP_TXn5
J48	VSS
J53	VSS
J55	VSS
J57	VSS
K6	VSS
K8	GBE_AUX_PWR_OK
K9	VSS
K46	VSS
K49	PCIE_EP_TXp3
K50	PCIE_EP_TXn3
K52	VSS
L2	SFP3_I2C_CLK
L5	EP_MAIN_PWR_OK
L12	EP_CRU_EN
L15	RSVD90
L17	VSS
L20	RSVD24
L22	VSS
L25	RSVD69
L27	RSVD72
L31	VSS
L33	PCIE_EP_TXn12
L36	VSS
L38	VSS
L41	VSS
L43	PCIE_EP_TXn6
L53	PCIE_EP_RXp1
L56	VSS
M1	VSS
M3	SRDS0_SD
M6	VSS
M8	GBE0_SWDP1
M9	SRDS3_SD
M11	VSS

Ball	Signal
M12	EP_SMBDAT
M15	EP_JTDO
M17	RSVD10
M20	RSVD15
M22	RSVD33
M25	RSVD70
M27	VSS
M31	PCIE_EP_TXn15
M33	VSS
M36	PCIE_EP_TXn11
M38	PCIE_EP_TXp8
M41	RSVD61
M43	VSS
M46	PCIE_EP_TXp1
M48	PCIE_EP_TXp2
M49	PCIE_EP_TXn2
M50	VSS
M52	VSS
M55	PCIE_EP_RXn1
M57	VSS
N2	SFP1_I2C_CLK
N4	SFP3_I2C_DATA
N6	GBE0_LED
N15	VSS
N52	VSS
N54	PCIE_EP_RXn0
N56	PCIE_EP_RXp0
P17	RSVD12
P20	VSS
P22	VCCAEPAUX
P25	RSVD71
P27	VSS
P31	PCIE_EP_TXp15
P33	PCIE_EP_TXn13
P36	PCIE_EP_TXp11
P38	PCIE_EP_TXn8
P41	VSS
P43	RSVD62
R2	GBE1_SWDP0
R4	GBE3_SWDP0
R6	GBE2_SWDP0

Ball	Signal
R8	SFP2_I2C_CLK
R9	VSS
R11	GBE1_LED
R12	GBE_EE_CS#
R14	RSVD88
R15	RSVD36
R17	VSS
R20	RSVD20
R22	VCCAEP1P8_CRU
R25	VSS
R27	VCCAEPAUX
R31	VSS
R33	PCIE_EP_TXp13
R36	VSS
R38	VSS
R41	PCIE_EP_CLK100P
R43	VSS
R44	VSS
R46	PCIE_EP_TXn1
R47	VSS
R49	PCIE_EP_TXn0
R50	PCIE_EP_TXp0
R52	VSS
R54	VSS
R56	VSS
T1	VSS
T3	SFP0_I2C_CLK
T6	VSS
T52	VSS
T55	DMI_RXp0
T57	VSS
U2	SFP0_I2C_DATA
U5	VSS
U8	SFP2_I2C_DATA
U9	GBE3_LED
U11	GBE_SMBDAT
U12	GBE_SMBALRT#
U14	VSS
U15	RSVD37
U17	VCCAEPAUX
U20	VCCAEPAUX

Ball	Signal
U22	VCCAEP1P8_CRU
U25	VCCEPAUX
U27	VSS
U31	VCCEP
U33	VSS
U36	VCCEP
U38	PCIE_EP_ICOMPI
U41	VSS
U43	PCIE_EP_CLK100N
U44	VSS
U46	VSS
U47	DMI_TXn0
U49	DMI_TXp0
U50	VSS
U53	VSS
U56	DMI_RXn0
V6	VSS
V20	VSS
V22	VCCEPAUX
V25	VSS
V27	VCCEP
V31	VSS
V33	VCCEP
V36	VSS
V38	VSS
V52	VSS
W1	VSS
W3	SRDSO_0_P
W5	VSS
W53	DMI_RXp1
W55	DMI_RXn1
W57	VSS
Y6	VSS
Y8	SFP1_I2C_DATA
Y9	GBE0_SWDP0
Y11	VSS
Y12	GBE_SMBCLK
Y14	RSVD87
Y15	EP_JTDI
Y17	VSS
Y18	VSS

Ball	Signal
Y20	VSS
Y22	VCCEPAUX
Y24	VSS
Y26	VSS
Y28	VCCEP
Y30	VCCEP
Y32	VSS
Y34	VCCEP
Y36	VSS
Y38	VCCAEP_PE
Y40	VSS
Y41	VCCAEP_PE
Y43	VCCAEP1P8_PE
Y44	PCIE_EP_VREF1P8
Y46	VSS
Y47	DMI_TXn1
Y49	DMI_TXp1
Y50	VSS
Y52	VSS
AA2	SRDSO_1_P
AA5	SRDSO_0_N
AA20	VSS
AA22	VCCEPAUX
AA24	VSS
AA26	VSS
AA28	VCCEP
AA30	VCCEP
AA32	VSS
AA34	VCCEP
AA36	VSS
AA38	VCCAEP_PE
AA53	VSS
AA56	DMI_RXp2
AB1	VSS
AB3	SRDSO_1_N
AB6	VSS
AB8	RSVD64
AB9	VSS
AB11	SRDSO_3_P
AB12	SRDSO_3_N
AB14	VSS

Ball	Signal
AB15	VCCPEP3P3AUX
AB17	VCCEPAUX
AB18	VSS
AB40	VSS
AB41	VCCAEP_PE
AB43	VCCAEP_PE
AB44	VSS
AB46	RSVD42
AB47	VSS
AB49	DMI_TXp2
AB50	DMI_TXn2
AB52	VSS
AB55	DMI_RXn2
AB57	VSS
AC2	SRDSO_2_P
AC4	SRDSO_2_N
AC6	VSS
AC20	VCCEPAUX
AC22	VSS
AC24	VCCEP1P0_CRU
AC26	VCCEP1P0_CRU
AC28	VSS
AC30	VCCEP
AC32	VSS
AC34	VCCEP
AC36	VSS
AC38	VCCAEP_PE
AC52	VSS
AC54	DMI_RXn3
AC56	DMI_RXp3
AE2	VSS
AE4	VSS
AE6	VSS
AE8	RSVD63
AE9	VSS
AE11	GBE_VREF1P8
AE12	VSS
AE14	VCCPEP3P3AUX
AE15	VSS
AE17	VCCPEP3P3AUX
AE18	VSS

Ball	Signal
AE20	VSS
AE22	VCCEPAUX
AE24	VCCEPAUX
AE26	VSS
AE28	VCCEP
AE30	VSS
AE32	VCCEP
AE34	VSS
AE36	VCCAEP_PE
AE38	VSS
AE40	VCCAEP_PE
AE41	VSS
AE43	VCCAEP_PE
AE44	VSS
AE46	RSVD41
AE47	DMI_IRCOMP
AE49	VSS
AE50	DMI_TXp3
AE52	VSS
AE54	DMI_CLK100P
AE56	DMI_CLK100N
AF1	GBE_CLK100P
AF3	GBE_CLK100N
AF6	VSS
AF52	DMI_TXn3
AF55	VSS
AF57	VSS
AG2	VSS
AG5	SRDSI_0_P
AG8	GBE_IRCOMP
AG9	VCCAEP1P8AUX
AG11	VCCAEP1P8AUX
AG12	VSS
AG14	VCCAEPAUX
AG15	VSS
AG17	VCCPEP3P3AUX
AG18	VSS
AG20	VCCEPAUX
AG22	VSS
AG24	VCCEPAUX
AG26	VCCEP

Ball	Signal
AG28	VSS
AG30	VCCEP
AG32	VSS
AG34	VCCEP
AG36	VSS
AG38	VCCAEP_PE
AG40	VSS
AG41	VCCAEP_PE
AG43	VCCAEP_PE
AG44	VCCA1P8_DMI
AG46	DMI_VREF
AG47	VSS
AG49	PCIE_RC_TXp0
AG50	PCIE_RC_TXn0
AG53	PCIE_RC_RXp0
AG56	PCIE_RC_Rxn0
AH6	VSS
AH52	VSS
AJ1	VSS
AJ3	SRDSI_1_P
AJ5	SRDSI_0_N
AJ20	VSS
AJ22	VCCEPAUX
AJ24	VCCEPAUX
AJ26	VSS
AJ28	VCCEP
AJ30	VSS
AJ32	VCCEP
AJ34	VSS
AJ36	VCCEP
AJ38	VSS
AJ53	PCIE_RC_RXp1
AJ55	PCIE_RC_Rxn1
AJ57	VSS
AK6	VSS
AK52	VSS
AL2	SRDSI_2_P
AL5	SRDSI_1_N
AL8	VCCAEP1P8AUX
AL9	VSS
AL11	VCCAEPAUX

Ball	Signal
AL12	VCCAEPAUX
AL14	VSS
AL15	VCCAEPAUX
AL17	VSS
AL18	VCCAEPAUX
AL20	VCCEPAUX
AL22	VSS
AL24	VSS
AL26	VCCEP
AL28	VSS
AL30	VCCEP
AL32	VSS
AL34	VCCEP
AL36	VSS
AL38	VCCA_DMI
AL40	VSS
AL41	VCCA_DMI
AL43	VSS
AL44	VCCA_DMI
AL46	VSS
AL47	PCIE_RC_TXp1
AL49	PCIE_RC_TXn1
AL50	VSS
AL53	VSS
AL56	PCIE_RC_RXp2
AM1	VSS
AM3	SRDSI_2_N
AM6	VSS
AM52	VSS
AM55	PCIE_RC_RXn2
AM57	VSS
AN2	SRDSI_3_P
AN4	SRDSI_3_N
AN6	VSS
AN8	VCC1P05
AN9	VCC1P05
AN11	VCCA1P8_TS1
AN12	VCCSUS1P8_USB
AN14	VSS
AN15	VCCA1P5_TS1
AN17	VCCA1P5_TS0

Ball	Signal
AN18	VSS
AN20	VSS
AN22	VSS
AN24	VCC
AN26	VSS
AN28	VCC
AN30	VSS
AN32	VCC
AN34	VCCA1P8_TS0
AN36	VSS
AN38	VCCA_DMI
AN40	VSS
AN41	VCCA_DMI
AN43	VSS
AN44	VCCA_DMI
AN46	PCIE_RC_TXp2
AN47	PCIE_RC_TXn2
AN49	VSS
AN50	PCIE_RC_TXp3
AN52	VSS
AN54	PCIE_RC_RXn3
AN56	PCIE_RC_RXp3
AR2	RSVD46
AR4	VSS
AR6	VSS
AR20	VSS
AR22	VCCSUS
AR24	VSS
AR26	VCC
AR28	VSS
AR30	VCC
AR32	VCC
AR34	VCC
AR36	VCC
AR38	VCCA_DMI
AR52	VSS
AR54	VSS
AR56	SATA4_RXP
AT1	VSS
AT3	RSVD45
AT6	VSS

Ball	Signal
AT8	VCCSUS3P3_USB
AT9	VSSA_USB
AT11	VCCASUS1P8_USB
AT12	VSS
AT14	VCCSUS1P8_USB
AT15	VCCSUS1P8_USB
AT17	VCCSUS1P8_USB
AT18	VSS
AT40	VCCA_DMI
AT41	VSS
AT43	VCCA_DMI
AT44	SATA_ICOMPO
AT46	VSS
AT47	SATA_VREF1P8
AT49	VSS
AT50	PCIE_RC_TXn3
AT52	VSS
AT55	SATA5_RXP
AT57	SATA4_RXN
AU2	USB_RBIASp
AU5	USB_RBIASn
AU20	VSS
AU22	VCCSUS
AU24	VSS
AU26	VCC
AU28	VSS
AU30	VCC
AU32	VSS
AU34	VCC
AU36	VSS
AU38	VSS
AU53	SATA5_RXN
AU56	VSS
AV6	VSS
AV8	VCCSUS_USB
AV9	VCCSUS3P3_USB
AV11	VCCSUS3P3_USB
AV12	VCCSUS3P3_USB
AV14	VCCSUS1P8
AV15	VSS
AV17	VCCSUS1P8

Ball	Signal
AV18	VSS
AV20	VCCSUS
AV22	VSS
AV24	VCCSUS
AV26	VCC3P3
AV28	VSS
AV30	VCC3P3
AV32	VSS
AV34	VCC3P3
AV36	VCC3P3
AV38	VCC3P3
AV40	VSS
AV41	VCCA_SATA
AV43	VSS
AV44	VCCA_SATA
AV46	VCCA1P8_SATA
AV47	SATA4_TXP
AV49	SATA4_TXN
AV50	VSS
AV52	VSS
AW1	VSS
AW3	USB_CLK96P
AW5	USB_CLK96N
AW53	SATA_CLK100N
AW55	SATA_CLK100P
AW57	VSS
AY6	VSS
AY20	VSS
AY22	VSS
AY25	VSS
AY27	VCC3P3
AY31	VSS
AY33	VCC3P3
AY36	VSS
AY38	VSS
AY52	VSS
BA2	USB0_Dn
BA5	USB0_Dp
BA8	VCCSUS_USB
BA9	VCCA1P8_USB
BA11	VSS

Ball	Signal
BA12	VCCSUS3P3_USB
BA14	VSS
BA15	VCCSUS3P3
BA17	VCCSUS1P8
BA20	VSS
BA22	VCCSUS
BA25	VCC3P3
BA27	VSS
BA31	VCC3P3
BA33	VSS
BA36	VCC3P3
BA38	VCC3P3
BA41	VSS
BA43	VCCA_SATA
BA44	RSVD44
BA46	RSVD43
BA47	VSS
BA49	SATA5_TXP
BA50	SATA5_TXN
BA53	PMSYNC
BA56	VSS
BB1	VSS
BB3	USB1_Dn
BB6	VSS
BB52	VSS
BB55	PECI
BB57	CPUPWRGD
BC2	USB1_Dp
BC4	USB2_Dn
BC6	VSS
BC8	VCCA1P8_USB
BC9	VSS
BC11	VCCA_USB
BC12	VCCA_USB
BC14	VCCSUS3P3
BC15	VCCSUS3P3
BC17	VSS
BC20	VCCSUS
BC22	VSS
BC25	VCC3P3
BC27	VCC1P8

Ball	Signal
BC31	VSS
BC33	VCC1P8
BC36	VCC1P8
BC38	VSS
BC41	VCCPCPU
BC43	VCCA_SATA
BC44	VSS
BC46	SATA5_GP/TEMP_ALERT #/GPIO49
BC47	SATA4_GP/GPIO16
BC49	RSVD47
BC50	GPIO20
BC52	VSS
BC54	SPI_MOSI
BC56	SPI_CS1#
BD15	VSS
BD17	VSS
BD20	VCCSUS
BD22	VCCSUS
BD25	VSS
BD27	VSS
BD31	VCC1P8
BD33	VSS
BD36	VSS
BD38	SATA_LED#
BD41	SPI_CLK
BD43	VSS
BE2	USB3_Dn
BE4	USB2_Dp
BE6	VSS
BE52	VSS
BE54	RSVD49
BE56	RSVD48
BF1	VSS
BF3	USB3_Dp
BF6	VSS
BF8	USB5_Dn
BF9	USB5_Dp
BF11	VSS
BF12	VSS
BF15	GPIO12
BF17	VCCSUS3P3

Ball	Signal
BF20	SLP_S5#/GPIO63
BF22	VSS
BF25	VCCSUS3P3_RTC
BF27	VCC3P3_RTC
BF31	VSS
BF33	LFRAME#
BF36	GPIO52
BF38	GPIO17
BF41	VSS
BF43	GPIO19
BF46	GPIO35
BF47	RCIN#
BF49	GPIO21
BF50	VSS
BF52	THRMTRIP#
BF55	RSVD50
BF57	VSS
BG2	USB4_Dn
BG5	VSS
BG12	PLTRST#
BG15	MST_SMBALERT#/GPIO11
BG17	GPIO28
BG20	PWRBTN#
BG22	GPIO10
BG25	RSVD39
BG27	VSS
BG31	SIU0_DCD#
BG33	LDRQ1#/GPIO23
BG36	VSS
BG38	GPIO32
BG41	VSS
BG43	GPIO50
BG46	VSS
BG53	ADR/GPIO37
BG56	GPIO18
BH6	VSS
BH8	RSVD74
BH9	TS1_IREF_N
BH49	VSS
BH50	SPI_MISO
BH52	VSS

Ball	Signal
BJ1	USB4_Dp
BJ3	VSS
BJ5	VSS
BJ10	GPIO15
BJ12	VSS
BJ15	RSVD78
BJ17	VSS
BJ20	SML1CLK/GPIO58
BJ22	VSS
BJ25	GPIO13
BJ27	RTCRST#
BJ31	SIU0_RXD
BJ33	LAD1
BJ36	LDRQ0#
BJ38	PCICLK
BJ41	BBS0
BJ43	GPIO4
BJ46	RSVD75
BJ48	SYS_RESET#
BJ53	REF_CLK14
BJ55	SPI_CS0#
BJ57	VSS
BK6	VSS
BK8	VSS
BK10	OC2#/GPIO41
BK12	OC0#/GPIO59
BK15	GPIO31
BK17	RI#
BK20	SML1DAT/GPIO75
BK22	GPIO43
BK25	GPIO25
BK27	PWROK
BK31	SRTCRST#
BK33	SIU0_RI#
BK36	SIU1_DCD#
BK38	VSS
BK41	SIU1_TXD
BK43	GPIO2
BK46	RSVD76
BK48	VSS
BK50	INIT3_3V#

Ball	Signal
BK52	VSS
BL2	MST_SMBCLK
BL4	MST_SMBDAT
BL54	A20GATE
BL56	SERIRQ
BM1	RSVD66
BM3	VSS
BM5	RSVD73
BM8	VSS
BM10	GPIO30
BM12	VSS
BM13	SUS_STAT#/GPIO61
BM15	VSS
BM16	VSS
BM18	OC3#/GPIO42
BM20	GPIO46
BM22	VSS
BM23	VSS
BM25	INTRUDER#
BM26	GPIO26
BM28	VSS
BM30	JTDI
BM32	VSS
BM33	SIU0_RTS#
BM35	VSS
BM36	SIU1_RXD
BM38	BBS1/GPIO51
BM40	VSS
BM42	GPIO53
BM43	VSS
BM45	GPIO5
BM46	VSS
BM48	VSS
BM50	VSS
BM53	NRBOOTS
BM55	VSS
BM57	VSS
BN2	RSVD68
BN4	OC1#/GPIO40
BN6	VSS
BN9	VSS

Ball	Signal
BN11	GPIO57
BN17	SLP_S4#
BN19	SLP_S3#
BN21	GPIO9
BN27	RSMRST#
BN29	VSS
BN31	JTMS
BN37	SIU1_RI#
BN39	UART_CLK
BN41	GPIO33
BN47	GPIO1
BN49	GPIO34
BN52	SIU1_DTR#
BN54	GPIO36
BN56	VSS
BP1	RSVD65
BP3	RSVD67
BP5	RSVD60
BP7	GPIO24
BP13	SUS_CLK/GPIO62
BP15	SYS_PWROK
BP23	GPIO14
BP25	IVCC_RTC
BP33	SIU0_CTS#
BP35	SIU1_DSR#
BP43	SIU0_DTR#
BP45	GPIO55
BP51	VSS
BP53	GPIO6
BP55	VSS
BP57	VSS
BR4	VSS
BR6	GPIO27
BR9	GPIO56
BR12	GPIO60
BR16	MEPWROK
BR19	VSS
BR22	GPIO73
BR26	RTCX1
BR29	JTCK
BR32	VSS

Ball	Signal
BR36	VSS
BR39	LAD0
BR42	GPIO3
BR46	GPIO54
BR49	SCLOCK/GPIO22
BR52	SDATAOUT0/GPIO39
BR54	VSS
BT2	VSS
BT5	RSVD57
BT7	VSS
BT11	VSS
BT13	GPIO72
BT15	VSS
BT17	DRAMPWRGD
BT21	GPIO45
BT23	GPIO44
BT25	RSVD53
BT27	VSS
BT31	RSVD38
BT33	RSVD77
BT35	SIU1_CTS#
BT37	WDT_TOUT#
BT41	LAD2
BT43	SIU0_TXD
BT45	BMBUSY#/GPIO0
BT47	GPIO7
BT51	SDATAOUT1/GPIO48
BT53	VSS
BT56	VSS
BU4	RSVD59
BU6	RSVD56
BU9	GPIO47
BU12	SML1ALERT#/GPIO74
BU16	GPIO8
BU19	WAKE#
BU22	VSS
BU26	RTCX2
BU29	JTDO
BU32	SIU0_DSR#
BU36	SIU1_RTS#
BU39	LAD3

Ball	Signal
BU42	VSS
BU46	VSS
BU49	SLOAD/GPIO38
BU52	VSS
BU54	VSS

Table 35-2. Alphabetical Signal Listing

Signal	Ball
A20GATE	BL54
GPIO4	BJ43
BBS0	BJ41
BBS1/GPIO51	BM38
BMBUSY#/GPIO0	BT45
CPUPWRGD	BB57
CRU_CLK100N	J22
CRU_CLK100P	J25
DMI_CLK100N	AE56
DMI_CLK100P	AE54
DMI_IRCOMP	AE47
DMI_RXn0	U56
DMI_RXn1	W55
DMI_RXn2	AB55
DMI_RXn3	AC54
DMI_RXp0	T55
DMI_RXp1	W53
DMI_RXp2	AA56
DMI_RXp3	AC56
DMI_TXn0	U47
DMI_TXn1	Y47
DMI_TXn2	AB50
DMI_TXn3	AF52
DMI_TXp0	U49
DMI_TXp1	Y49
DMI_TXp2	AB49
DMI_TXp3	AE50
DMI_VREF	AG46
DRAMPWRGD	BT17
EP_CRU_EN	L12
EP_JTCK	A12
EP_JTDI	Y15
EP_JTDO	M15
EP_JTMS	J17
EP_JTRST#	D13
EP_MAIN_PWR_OK	L5
EP_SMBALRT#	C12
EP_SMBCLK	B13
EP_SMBDAT	M12
GBE_AUX_PWR_AVAIL	B7
GBE_AUX_PWR_OK	K8
GBE_CLK100N	AF3
GBE_CLK100P	AF1
GBE_EE_CS#	R12
GBE_EE_DI	D5
GBE_EE_DO	F8
GBE_EE_SK	J10
GBE_IRCOMP	AG8
GBE_SMBALRT#	U12
GBE_SMBCLK	Y12
GBE_SMBDAT	U11
GBE_VREF1P8	AE11
GBE_WAKE#	H12
GBE0_LED	N6
GBE0_SWDP0	Y9
GBE0_SWDP1	M8
GBE1_LED	R11
GBE1_SWDP0	R2
GBE1_SWDP1	J3
GBE2_LED	E4
GBE2_SWDP0	R6
GBE2_SWDP1	G4
GBE3_LED	U9
GBE3_SWDP0	R4
GBE3_SWDP1	H10
GPIO1	BN47
GPIO2	BK43
GPIO3	BR42
GPIO5	BM45
GPIO6	BP53
GPIO7	BT47
GPIO8	BU16
GPIO9	BN21
GPIO10	BG22
GPIO12	BF15
GPIO13	BJ25
GPIO14	BP23
GPIO15	BJ10
GPIO17	BF38
GPIO18	BG56
GPIO19	BF43
GPIO20	BC50
GPIO21	BF49
GPIO24	BP7
GPIO25	BK25
GPIO26	BM26
GPIO27	BR6
GPIO28	BG17
GPIO30	BM10
GPIO31	BK15
GPIO32	BG38
GPIO33	BN41
GPIO34	BN49
GPIO35	BF46
GPIO36	BN54
ADR/GPIO37	BG53
GPIO43	BK22
GPIO44	BT23
GPIO45	BT21
GPIO46	BM20
GPIO47	BU9
GPIO50	BG43
GPIO52	BF36
GPIO53	BM42
GPIO54	BR46
GPIO55	BP45
GPIO56	BR9
GPIO57	BN11
GPIO60	BR12
GPIO72	BT13
GPIO73	BR22
INIT3_3V#	BK50
INTRUDER#	BM25
IVCC_RTC	BP25
JTCK	BR29
JTDI	BM30
JTDO	BU29
JTMS	BN31
LAD0	BR39
LAD1	BJ33
LAD2	BT41
LAD3	BU39
LDRQ0#	BJ36

Signal	Ball
LDRQ1#/GPIO23	BG33
LFRAME#	BF33
MEPWROK	BR16
MST_SMBALERT#/GPIO 11	BG15
MST_SMBCLK	BL2
MST_SMBDAT	BL4
NRBOOTS	BM53
OC0#/GPIO59	BK12
OC1#/GPIO40	BN4
OC2#/GPIO41	BK10
OC3#/GPIO42	BM18
PCICLK	BJ38
PCIE_EP_CLK100N	U43
PCIE_EP_CLK100P	R41
PCIE_EP_ICOMPI	U38
PCIE_EP_RST#	F10
PCIE_EP_RXn0	N54
PCIE_EP_RXn1	M55
PCIE_EP_RXn2	G54
PCIE_EP_RXn3	B51
PCIE_EP_RXn4	B47
PCIE_EP_RXn5	B45
PCIE_EP_RXn6	D43
PCIE_EP_RXn7	C42
PCIE_EP_RXn8	E41
PCIE_EP_RXn9	C39
PCIE_EP_RXn10	B37
PCIE_EP_RXn11	B35
PCIE_EP_RXn12	D33
PCIE_EP_RXn13	E31
PCIE_EP_RXn14	E29
PCIE_EP_RXn15	E27
PCIE_EP_RXp0	N56
PCIE_EP_RXp1	L53
PCIE_EP_RXp2	G56
PCIE_EP_RXp3	D51
PCIE_EP_RXp4	C46
PCIE_EP_RXp5	D45
PCIE_EP_RXp6	B43
PCIE_EP_RXp7	B41
PCIE_EP_RXp8	E39

Signal	Ball
PCIE_EP_RXp9	E37
PCIE_EP_RXp10	C36
PCIE_EP_RXp11	D35
PCIE_EP_RXp12	B33
PCIE_EP_RXp13	B31
PCIE_EP_RXp14	C29
PCIE_EP_RXp15	B27
PCIE_EP_TXn0	R49
PCIE_EP_TXn1	R46
PCIE_EP_TXn2	M49
PCIE_EP_TXn3	K50
PCIE_EP_TXn4	H50
PCIE_EP_TXn5	J46
PCIE_EP_TXn6	L43
PCIE_EP_TXn7	J41
PCIE_EP_TXn8	P38
PCIE_EP_TXn9	H38
PCIE_EP_TXn10	H36
PCIE_EP_TXn11	M36
PCIE_EP_TXn12	L33
PCIE_EP_TXn13	P33
PCIE_EP_TXn14	H31
PCIE_EP_TXn15	M31
PCIE_EP_TXp0	R50
PCIE_EP_TXp1	M46
PCIE_EP_TXp2	M48
PCIE_EP_TXp3	K49
PCIE_EP_TXp4	H48
PCIE_EP_TXp5	H46
PCIE_EP_TXp6	J43
PCIE_EP_TXp7	H41
PCIE_EP_TXp8	M38
PCIE_EP_TXp9	J38
PCIE_EP_TXp10	J36
PCIE_EP_TXp11	P36
PCIE_EP_TXp12	J33
PCIE_EP_TXp13	R33
PCIE_EP_TXp14	J31
PCIE_EP_TXp15	P31
PCIE_EP_VREF1P8	Y44
PCIE_RC_Rxn0	AG56

Signal	Ball
PCIE_RC_Rxn1	AJ55
PCIE_RC_RXn2	AM55
PCIE_RC_RXn3	AN54
PCIE_RC_RXp0	AG53
PCIE_RC_RXp1	AJ53
PCIE_RC_RXp2	AL56
PCIE_RC_RXp3	AN56
PCIE_RC_TXn0	AG50
PCIE_RC_TXn1	AL49
PCIE_RC_TXn2	AN47
PCIE_RC_TXn3	AT50
PCIE_RC_TXp0	AG49
PCIE_RC_TXp1	AL47
PCIE_RC_TXp2	AN46
PCIE_RC_TXp3	AN50
PECI	BB55
PLTRST#	BG12
PMSYNC	BA53
PWRBTN#	BG20
PWROK	BK27
RCIN#	BF47
REF_CLK14	BJ53
RI#	BK17
RSMRST#	BN27
RSVD0	E2
RSVD1	B5
RSVD2	B2
RSVD3	F1
RSVD4	D1
RSVD5	B15
RSVD6	C4
RSVD7	H17
RSVD8	A6
RSVD9	F18
RSVD10	M17
RSVD11	B17
RSVD12	P17
RSVD13	C16
RSVD14	J20
RSVD15	M20
RSVD16	F13

Signal	Ball
RSVD17	H22
RSVD18	F22
RSVD19	F23
RSVD20	R20
RSVD21	B21
RSVD22	D15
RSVD23	D23
RSVD24	L20
RSVD25	B23
RSVD26	E17
RSVD27	A22
RSVD28	C19
RSVD29	B25
RSVD30	F20
RSVD31	D25
RSVD32	H27
RSVD33	M22
RSVD34	E21
RSVD35	H25
RSVD36	R15
RSVD37	U15
RSVD38	BT31
RSVD39	BG25
RSVD41	AE46
RSVD42	AB46
RSVD43	BA46
RSVD44	BA44
RSVD45	AT3
RSVD46	AR2
RSVD47	BC49
RSVD48	BE56
RSVD49	BE54
RSVD50	BF55
RSVD51	D55
RSVD52	C54
RSVD53	BT25
RSVD54	E56
RSVD55	D57
RSVD56	BU6
RSVD57	BT5
RSVD59	BU4

Signal	Ball
RSVD60	BP5
RSVD61	M41
RSVD62	P43
RSVD63	AE8
RSVD64	AB8
RSVD65	BP1
RSVD66	BM1
RSVD67	BP3
RSVD68	BN2
RSVD69	L25
RSVD70	M25
RSVD71	P25
RSVD72	L27
RSVD73	BM5
RSVD74	BH8
RSVD75	BJ46
RSVD76	BK46
RSVD77	BT33
RSVD78	BJ15
RSVD79	C9
RSVD80	E11
RSVD81	B11
RSVD82	E9
RSVD83	J1
RSVD84	H6
RSVD85	F5
RSVD86	H8
RSVD87	Y14
RSVD88	R14
RSVD89	J15
RSVD90	L15
RTCRST#	BJ27
RTCX1	BR26
RTCX2	BU26
SATA_CLK100N	AW53
SATA_CLK100P	AW55
SATA_ICOMPO	AT44
SATA_LED#	BD38
SATA_VREF1P8	AT47
SATA4_GP/GPIO16	BC47
SATA4_RXN	AT57

Signal	Ball
SATA4_RXP	AR56
SATA4_TXN	AV49
SATA4_TXP	AV47
SATA5_GP/TEMP_ALERT #/GPIO49	BC46
SATA5_RXN	AU53
SATA5_RXP	AT55
SATA5_TXN	BA50
SATA5_TXP	BA49
SCLOCK/GPIO22	BR49
SDATAOUT0/GPIO39	BR52
SDATAOUT1/GPIO48	BT51
SERIRQ	BL56
SFP0_I2C_CLK	T3
SFP0_I2C_DATA	U2
SFP1_I2C_CLK	N2
SFP1_I2C_DATA	Y8
SFP2_I2C_CLK	R8
SFP2_I2C_DATA	U8
SFP3_I2C_CLK	L2
SFP3_I2C_DATA	N4
SIU0_CTS#	BP33
SIU0_DCD#	BG31
SIU0_DSR#	BU32
SIU0_DTR#	BP43
SIU0_RI#	BK33
SIU0_RTS#	BM33
SIU0_RXD	BJ31
SIU0_TXD	BT43
SIU1_CTS#	BT35
SIU1_DCD#	BK36
SIU1_DSR#	BP35
SIU1_DTR#	BN52
SIU1_RI#	BN37
SIU1_RTS#	BU36
SIU1_RXD	BM36
SIU1_TXD	BK41
SLOAD/GPIO38	BU49
SLP_S3#	BN19
SLP_S4#	BN17
SLP_S5#/GPIO63	BF20
SML1ALERT#/GPIO74	BU12

Signal	Ball
SML1CLK/GPIO58	BJ20
SML1DAT/GPIO75	BK20
SPI_CLK	BD41
SPI_CS0#	BJ55
SPI_CS1#	BC56
SPI_MISO	BH50
SPI_MOSI	BC54
SRDS0_SD	M3
SRDS1_SD	C6
SRDS2_SD	F3
SRDS3_SD	M9
SRDSI_0_N	AJ5
SRDSI_0_P	AG5
SRDSI_1_N	AL5
SRDSI_1_P	AJ3
SRDSI_2_N	AM3
SRDSI_2_P	AL2
SRDSI_3_N	AN4
SRDSI_3_P	AN2
SRDSO_0_N	AA5
SRDSO_0_P	W3
SRDSO_1_N	AB3
SRDSO_1_P	AA2
SRDSO_2_N	AC4
SRDSO_2_P	AC2
SRDSO_3_N	AB12
SRDSO_3_P	AB11
SRTCRST#	BK31
SUS_CLK/GPIO62	BP13
SUS_STAT#/GPIO61	BM13
SYS_PWROK	BP15
SYS_RESET#	BJ48
THRMTRIP#	BF52
TS0_IREF_N	D53
TS1_IREF_N	BH9
UART_CLK	BN39
USB_CLK96N	AW5
USB_CLK96P	AW3
USB_RBIASn	AU5
USB_RBIASp	AU2
USB0_Dn	BA2

Signal	Ball
USB0_Dp	BA5
USB1_Dn	BB3
USB1_Dp	BC2
USB2_Dn	BC4
USB2_Dp	BE4
USB3_Dn	BE2
USB3_Dp	BF3
USB4_Dn	BG2
USB4_Dp	BJ1
USB5_Dn	BF8
USB5_Dp	BF9
VCC	AN24
VCC	AN28
VCC	AN32
VCC	AR26
VCC	AR30
VCC	AR32
VCC	AR34
VCC	AR36
VCC	AU26
VCC	AU30
VCC	AU34
VCC1P05	AN8
VCC1P05	AN9
VCC1P8	BC27
VCC1P8	BC33
VCC1P8	BC36
VCC1P8	BD31
VCC3P3	AV26
VCC3P3	AV30
VCC3P3	AV34
VCC3P3	AV36
VCC3P3	AV38
VCC3P3	AY27
VCC3P3	AY33
VCC3P3	BA25
VCC3P3	BA31
VCC3P3	BA36
VCC3P3	BA38
VCC3P3	BC25
VCC3P3_RTC	BF27

Signal	Ball
VCCA_DMI	AL38
VCCA_DMI	AL41
VCCA_DMI	AL44
VCCA_DMI	AN38
VCCA_DMI	AN41
VCCA_DMI	AN44
VCCA_DMI	AR38
VCCA_DMI	AT40
VCCA_DMI	AT43
VCCA_SATA	AV41
VCCA_SATA	AV44
VCCA_SATA	BA43
VCCA_SATA	BC43
VCCA_USB	BC11
VCCA_USB	BC12
VCCA1P5_TS0	AN17
VCCA1P5_TS1	AN15
VCCA1P8_DMI	AG44
VCCA1P8_SATA	AV46
VCCA1P8_TS0	AN34
VCCA1P8_TS1	AN11
VCCA1P8_USB	BA9
VCCA1P8_USB	BC8
VCCAEP_PE	Y38
VCCAEP_PE	Y41
VCCAEP_PE	AA38
VCCAEP_PE	AB41
VCCAEP_PE	AB43
VCCAEP_PE	AC38
VCCAEP_PE	AE36
VCCAEP_PE	AE40
VCCAEP_PE	AE43
VCCAEP_PE	AG38
VCCAEP_PE	AG41
VCCAEP_PE	AG43
VCCAEP1P8_CRU	R22
VCCAEP1P8_CRU	U22
VCCAEP1P8_PE	Y43
VCCAEP1P8AUX	AG9
VCCAEP1P8AUX	AG11
VCCAEP1P8AUX	AL8

Signal	Ball
VCCAEPAUX	P22
VCCAEPAUX	R27
VCCAEPAUX	U17
VCCAEPAUX	U20
VCCAEPAUX	AG14
VCCAEPAUX	AL11
VCCAEPAUX	AL12
VCCAEPAUX	AL15
VCCAEPAUX	AL18
VCCASUS1P8_USB	AT11
VCCEP	U31
VCCEP	U36
VCCEP	V27
VCCEP	V33
VCCEP	Y28
VCCEP	Y30
VCCEP	Y34
VCCEP	AA28
VCCEP	AA30
VCCEP	AA34
VCCEP	AC30
VCCEP	AC34
VCCEP	AE28
VCCEP	AE32
VCCEP	AG26
VCCEP	AG30
VCCEP	AG34
VCCEP	AJ28
VCCEP	AJ32
VCCEP	AJ36
VCCEP	AL26
VCCEP	AL30
VCCEP	AL34
VCCEP1P0_CRU	AC24
VCCEP1P0_CRU	AC26
VCCEPAUX	U25
VCCEPAUX	V22
VCCEPAUX	Y22
VCCEPAUX	AA22
VCCEPAUX	AB17
VCCEPAUX	AC20

Signal	Ball
VCCEPAUX	AE22
VCCEPAUX	AE24
VCCEPAUX	AG20
VCCEPAUX	AG24
VCCEPAUX	AJ22
VCCEPAUX	AJ24
VCCEPAUX	AL20
VCCPCPU	BC41
VCCPEP3P3AUX	AB15
VCCPEP3P3AUX	AE14
VCCPEP3P3AUX	AE17
VCCPEP3P3AUX	AG17
VCCSUS	AR22
VCCSUS	AU22
VCCSUS	AV20
VCCSUS	AV24
VCCSUS	BA22
VCCSUS	BC20
VCCSUS	BD20
VCCSUS	BD22
VCCSUS_USB	AV8
VCCSUS_USB	BA8
VCCSUS1P8	AV14
VCCSUS1P8	AV17
VCCSUS1P8	BA17
VCCSUS1P8_USB	AN12
VCCSUS1P8_USB	AT14
VCCSUS1P8_USB	AT15
VCCSUS1P8_USB	AT17
VCCSUS3P3	BA15
VCCSUS3P3	BC14
VCCSUS3P3	BC15
VCCSUS3P3	BF17
VCCSUS3P3_RTC	BF25
VCCSUS3P3_USB	AT8
VCCSUS3P3_USB	AV9
VCCSUS3P3_USB	AV11
VCCSUS3P3_USB	AV12
VCCSUS3P3_USB	BA12
VSS	A4
VSS	A9

Signal	Ball
VSS	A16
VSS	A19
VSS	A26
VSS	A29
VSS	A32
VSS	A36
VSS	A39
VSS	A42
VSS	A46
VSS	A49
VSS	A52
VSS	A54
VSS	B53
VSS	B56
VSS	C22
VSS	C26
VSS	C32
VSS	C49
VSS	C52
VSS	D3
VSS	D7
VSS	D47
VSS	E6
VSS	E19
VSS	E49
VSS	E52
VSS	E54
VSS	F12
VSS	F15
VSS	F16
VSS	F25
VSS	F26
VSS	F28
VSS	F30
VSS	F32
VSS	F33
VSS	F35
VSS	F36
VSS	F38
VSS	F40
VSS	F42

Signal	Ball
VSS	F43
VSS	F45
VSS	F46
VSS	F48
VSS	F50
VSS	F53
VSS	F55
VSS	F57
VSS	G2
VSS	H15
VSS	H20
VSS	H33
VSS	H43
VSS	H52
VSS	J5
VSS	J12
VSS	J27
VSS	J48
VSS	J53
VSS	J55
VSS	J57
VSS	K6
VSS	K9
VSS	K46
VSS	K52
VSS	L17
VSS	L22
VSS	L31
VSS	L36
VSS	L38
VSS	L41
VSS	L56
VSS	M1
VSS	M6
VSS	M11
VSS	M27
VSS	M33
VSS	M43
VSS	M50
VSS	M52
VSS	M57

Signal	Ball
VSS	N15
VSS	N52
VSS	P20
VSS	P27
VSS	P41
VSS	R9
VSS	R17
VSS	R25
VSS	R31
VSS	R36
VSS	R38
VSS	R43
VSS	R44
VSS	R47
VSS	R52
VSS	R54
VSS	R56
VSS	T1
VSS	T6
VSS	T52
VSS	T57
VSS	U5
VSS	U14
VSS	U27
VSS	U33
VSS	U41
VSS	U44
VSS	U46
VSS	U50
VSS	U53
VSS	V6
VSS	V20
VSS	V25
VSS	V31
VSS	V36
VSS	V38
VSS	V52
VSS	W1
VSS	W5
VSS	W57
VSS	Y6

Signal	Ball
VSS	Y11
VSS	Y17
VSS	Y18
VSS	Y20
VSS	Y24
VSS	Y26
VSS	Y32
VSS	Y36
VSS	Y40
VSS	Y46
VSS	Y50
VSS	Y52
VSS	AA20
VSS	AA24
VSS	AA26
VSS	AA32
VSS	AA36
VSS	AA53
VSS	AB1
VSS	AB6
VSS	AB9
VSS	AB14
VSS	AB18
VSS	AB40
VSS	AB44
VSS	AB47
VSS	AB52
VSS	AB57
VSS	AC6
VSS	AC22
VSS	AC28
VSS	AC32
VSS	AC36
VSS	AC52
VSS	AE2
VSS	AE4
VSS	AE6
VSS	AE9
VSS	AE12
VSS	AE15
VSS	AE18

Signal	Ball
VSS	AE20
VSS	AE26
VSS	AE30
VSS	AE34
VSS	AE38
VSS	AE41
VSS	AE44
VSS	AE49
VSS	AE52
VSS	AF6
VSS	AF55
VSS	AF57
VSS	AG2
VSS	AG12
VSS	AG15
VSS	AG18
VSS	AG22
VSS	AG28
VSS	AG32
VSS	AG36
VSS	AG40
VSS	AG47
VSS	AH6
VSS	AH52
VSS	AJ1
VSS	AJ20
VSS	AJ26
VSS	AJ30
VSS	AJ34
VSS	AJ38
VSS	AJ57
VSS	AK6
VSS	AK52
VSS	AL9
VSS	AL14
VSS	AL17
VSS	AL22
VSS	AL24
VSS	AL28
VSS	AL32
VSS	AL36

Signal	Ball
VSS	AL40
VSS	AL43
VSS	AL46
VSS	AL50
VSS	AL53
VSS	AM1
VSS	AM6
VSS	AM52
VSS	AM57
VSS	AN6
VSS	AN14
VSS	AN18
VSS	AN20
VSS	AN22
VSS	AN26
VSS	AN30
VSS	AN36
VSS	AN40
VSS	AN43
VSS	AN49
VSS	AN52
VSS	AR4
VSS	AR6
VSS	AR20
VSS	AR24
VSS	AR28
VSS	AR52
VSS	AR54
VSS	AT1
VSS	AT6
VSS	AT12
VSS	AT18
VSS	AT41
VSS	AT46
VSS	AT49
VSS	AT52
VSS	AU20
VSS	AU24
VSS	AU28
VSS	AU32
VSS	AU36

Signal	Ball
VSS	AU38
VSS	AU56
VSS	AV6
VSS	AV15
VSS	AV18
VSS	AV22
VSS	AV28
VSS	AV32
VSS	AV40
VSS	AV43
VSS	AV50
VSS	AV52
VSS	AW1
VSS	AW57
VSS	AY6
VSS	AY20
VSS	AY22
VSS	AY25
VSS	AY31
VSS	AY36
VSS	AY38
VSS	AY52
VSS	BA11
VSS	BA14
VSS	BA20
VSS	BA27
VSS	BA33
VSS	BA41
VSS	BA47
VSS	BA56
VSS	BB1
VSS	BB6
VSS	BB52
VSS	BC6
VSS	BC9
VSS	BC17
VSS	BC22
VSS	BC31
VSS	BC38
VSS	BC44
VSS	BC52

Signal	Ball
VSS	BD15
VSS	BD17
VSS	BD25
VSS	BD27
VSS	BD33
VSS	BD36
VSS	BD43
VSS	BE6
VSS	BE52
VSS	BF1
VSS	BF6
VSS	BF11
VSS	BF12
VSS	BF22
VSS	BF31
VSS	BF41
VSS	BF50
VSS	BF57
VSS	BG5
VSS	BG27
VSS	BG36
VSS	BG41
VSS	BG46
VSS	BH6
VSS	BH49
VSS	BH52
VSS	BJ3
VSS	BJ5
VSS	BJ12
VSS	BJ17
VSS	BJ22
VSS	BJ57
VSS	BK6
VSS	BK8
VSS	BK38
VSS	BK48
VSS	BK52
VSS	BM3
VSS	BM8
VSS	BM12
VSS	BM15

Signal	Ball
VSS	BM16
VSS	BM22
VSS	BM23
VSS	BM28
VSS	BM32
VSS	BM35
VSS	BM40
VSS	BM43
VSS	BM46
VSS	BM48
VSS	BM50
VSS	BM55
VSS	BM57
VSS	BN6
VSS	BN9
VSS	BN29
VSS	BN56
VSS	BP51
VSS	BP55
VSS	BP57
VSS	BR4
VSS	BR19
VSS	BR32
VSS	BR36
VSS	BR54
VSS	BT2
VSS	BT7
VSS	BT11
VSS	BT15
VSS	BT27
VSS	BT53
VSS	BT56
VSS	BU22
VSS	BU42
VSS	BU46
VSS	BU52
VSS	BU54
VSSA_USB	AT9
WAKE#	BU19
WDT_TOUT#	BT37

35.2.2 Ball Map Illustrations

Table 35-3, "Ball Map Color Code" contains the signal color codes used in Table 35-4 through Table 35-7.

Table 35-3. Ball Map Color Code

Color	Signals
	USB; SATA; SIU, PCIE_EP, SFP
	VSSA; VSS
	RSVD
	PCIE_RC
	GbE; SRD
	GPIO
	SPI
	DMI
	VCCEP; VCCAEP; VCCPEP
	VCC; VCCA_; VCC3P3; VCC1P8; VCCA1
	VCCSUS

Table 35-4. Bottom View Left (Sheet 1 of 4)

	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	T	U	V	W	Y	AA	AB	AC	AE	AF	AG	AH	AJ
1				RSVD 4		RSVD 3			RSVD 83			VSS				VSS			VSS			VSS			GBE_ CLK10 0P			VSS
2		RSVD 2			RSVD 0		VSS				SFP3_ I2C_C LK		SFP1_ I2C_C LK		GBE1 _SWD P0		SFP0_ I2C_D ATA				SRDS O_1_ P		SRDS O_2_ P	VSS		VSS		
3				VSS		SRDS 2_SD			GBE1 _SWD P1			SRDS 0_SD				SFP0_ I2C_C LK			SRDS O_0_ P			SRDS O_1_ N			GBE_ CLK10 0N			SRDSI _1_P
4	VSS		RSVD 6		GBE2 _LED		GBE2 _SWD P1						SFP3_ I2C_D ATA		GBE3 _SWD P0								SRDS O_2_ N	VSS				
5		RSVD 1		GBE_ EE_DI		RSVD 85			VSS		EP_M AIN_P WR_O K						VSS		VSS		SRDS O_0_ N					SRDSI _0_P		SRDSI _0_N
6	RSVD 8		SRDS 1_SD		VSS			RSVD 84		VSS		VSS	GBE0 _LED		GBE2 _SWD P0	VSS		VSS		VSS		VSS	VSS	VSS	VSS		VSS	
7		GBE_ AUX_ PWR_ AVAIL		VSS																								
8						GBE_ EE_D O		RSVD 86		GBE_ AUX_ PWR_ OK		GBE0 _SWD P1			SFP2_ I2C_C LK		SFP2_ I2C_D ATA			SFP1_ I2C_D ATA		RSVD 64		RSVD 63		GBE_I RCOM P		
9	VSS		RSVD 79		RSVD 82					VSS		SRDS 3_SD			VSS		GBE3 _LED			GBE0 _SWD P0		VSS		VSS		VCCA EP1P8 AUX		

Table 35-4. Bottom View Left (Sheet 2 of 4)

	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	T	U	V	W	Y	AA	AB	AC	AE	AF	AG	AH	AJ
10						PCIE_EP_RST#		GBE3_SWDP1	GBE_EE_SK																			
11		RSVD 81			RSVD 80							VSS			GBE1_LED		GBE_SMBDAT			VSS		SRDSO_3_P		GBE_VREF1P8		VCCAEP1P8AUX		
12	EP_JTCK		EP_SMBALRT#			VSS		GBE_WAKE#	VSS		EP_CRU_EN	EP_SMBDAT			GBE_EE_CS#		GBE_SMBALRT#			GBE_SMBCLK		SRDSO_3_N		VSS		VSS		
13		EP_SMBCLK		EP_JTRST#		RSVD 16																						
14															RSVD 88		VSS			RSVD 87		VSS		VCCPEP3P3AUX		VCCAEPAUX		
15		RSVD 5		RSVD 22		VSS		VSS	RSVD 89		RSVD 90	EP_JTDO	VSS		RSVD 36		RSVD 37			EP_JTDI		VCCPEP3P3AUX		VSS		VSS		
16	VSS		RSVD 13			VSS																						
17		RSVD 11			RSVD 26			RSVD 7	EP_JTMS		VSS	RSVD 10		RSVD 12	VSS		VCCAEPAUX			VSS		VCCEPAUX		VCCPEP3P3AUX		VCCPEP3P3AUX		
18						RSVD 9														VSS		VSS		VSS		VSS		
19	VSS		RSVD 28		VSS																							
20						RSVD 30		VSS	RSVD 14		RSVD 24	RSVD 15		VSS	RSVD 20		VCCAEPAUX	VSS		VSS	VSS		VCCEPAUX	VSS		VCCEPAUX		VSS
21		RSVD 21			RSVD 34																							
22	RSVD 27		VSS			RSVD 18		RSVD 17	CRU_CLK100N		VSS	RSVD 33		VCCAEPAUX	VCCAEP1P8_CRU		VCCAEP1P8_CRU	VCCEPAUX		VCCEPAUX	VCCEPAUX		VSS	VCCEPAUX		VSS		VCCEPAUX
23		RSVD 25		RSVD 23		RSVD 19																						
24																				VSS	VSS		VCCEP1P0_CRU	VCCEPAUX		VCCEPAUX		VCCEPAUX
25		RSVD 29		RSVD 31		VSS		RSVD 35	CRU_CLK100P		RSVD 69	RSVD 70		RSVD 71	VSS		VCCEPAUX	VSS										
26	VSS		VSS			VSS														VSS	VSS		VCCEP1P0_CRU	VSS		VCCEP		VSS
27		PCIE_EP_RXp15			PCIE_EP_RXn15			RSVD 32	VSS		RSVD 72	VSS		VSS	VCCAEPAUX		VSS	VCCEP										

Table 35-4. Bottom View Left (Sheet 3 of 4)

	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	T	U	V	W	Y	AA	AB	AC	AE	AF	AG	AH	AJ
28						VSS														VCCEP	VCCEP		VSS	VCCEP		VSS		VCCEP
29	VSS		PCIE_EP_RXp14		PCIE_EP_RXn14																							
30						VSS														VCCEP	VCCEP		VCCEP	VSS		VCCEP		VSS
31		PCIE_EP_RXp13			PCIE_EP_RXn13			PCIE_EP_TXn14	PCIE_EP_TXp14		VSS	PCIE_EP_TXn15		PCIE_EP_TXp15	VSS		VCCEP	VSS										
32	VSS		VSS			VSS														VSS	VSS		VSS	VCCEP		VSS		VCCEP
33		PCIE_EP_RXp12		PCIE_EP_RXn12		VSS		VSS	PCIE_EP_TXp12		PCIE_EP_TXn12	VSS		PCIE_EP_TXn13	PCIE_EP_TXp13		VSS	VCCEP										
34																				VCCEP	VCCEP		VCCEP	VSS		VCCEP		VSS
35		PCIE_EP_RXn11		PCIE_EP_RXp11		VSS																						
36	VSS		PCIE_EP_RXp10			VSS		PCIE_EP_TXn10	PCIE_EP_TXp10		VSS	PCIE_EP_TXn11		PCIE_EP_TXp11	VSS		VCCEP	VSS		VSS	VSS		VSS	VCCAEP_PE		VSS		VCCEP
37		PCIE_EP_RXn10			PCIE_EP_RXp9																							
38						VSS		PCIE_EP_TXn9	PCIE_EP_TXp9		VSS	PCIE_EP_TXp8		PCIE_EP_TXn8	VSS		PCIE_EP_ICOMPI	VSS		VCCAEP_PE	VCCAEP_PE		VCCAEP_PE	VSS		VCCAEP_PE		VSS
39	VSS		PCIE_EP_RXn9		PCIE_EP_RXp8																							
40						VSS														VSS		VSS		VCCAEP_PE		VSS		
41		PCIE_EP_RXp7			PCIE_EP_RXn8			PCIE_EP_TXp7	PCIE_EP_TXn7		VSS	RSVD61		VSS	PCIE_EP_CLK100P		VSS			VCCAEP_PE		VCCAEP_PE		VSS		VCCAEP_PE		
42	VSS		PCIE_EP_RXn7			VSS																						
43		PCIE_EP_RXp6		PCIE_EP_RXn6		VSS		VSS	PCIE_EP_TXp6		PCIE_EP_TXn6	VSS		RSVD62	VSS		PCIE_EP_CLK100N			VCCAEP1P8_PE		VCCAEP_PE		VCCAEP_PE		VCCAEP_PE		
44															VSS		VSS			PCIE_EP_VREF1P8		VSS		VSS		VCCA1P8_DMI		
45		PCIE_EP_RXn5		PCIE_EP_RXp5		VSS																						

Table 35-4. Bottom View Left (Sheet 4 of 4)

	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	T	U	V	W	Y	AA	AB	AC	AE	AF	AG	AH	AJ
46	VSS		PCIE_EP_RXp4			VSS		PCIE_EP_TXp5	PCIE_EP_TXn5	VSS		PCIE_EP_TXp1			PCIE_EP_TXn1		VSS			VSS		RSVD 42		RSVD 41		DMI_VREF		
47		PCIE_EP_RXn4		VSS											VSS		DMI_TXn0			DMI_TXn1		VSS		DMI_IRCOMP		VSS		
48						VSS		PCIE_EP_TXp4	VSS			PCIE_EP_TXp2																
49	VSS		VSS		VSS					PCIE_EP_TXp3		PCIE_EP_TXn2			PCIE_EP_TXn0		DMI_TXp0			DMI_TXp1		DMI_TXp2		VSS		PCIE_RC_TXp0		
50						VSS		PCIE_EP_TXn4		PCIE_EP_TXn3		VSS			PCIE_EP_TXp0		VSS			VSS		DMI_TXn2		DMI_TXp3		PCIE_RC_TXn0		
51		PCIE_EP_RXn3		PCIE_EP_RXp3																								
52	VSS		VSS		VSS			VSS		VSS		VSS	VSS		VSS	VSS		VSS		VSS		VSS	VSS	VSS	DMI_TXn3		VSS	
53		VSS		TS0_IREF_N		VSS			VSS		PCIE_EP_RXp1						VSS		DMI_RXp1		VSS					PCIE_RC_RXp0		PCIE_RC_RXp1
54	VSS		RSVD 52		VSS		PCIE_EP_RXn2						PCIE_EP_RXn0		VSS								DMI_RXn3	DMI_CLK100P				
55				RSVD 51		VSS			VSS			PCIE_EP_RXn1				DMI_RXp0			DMI_RXn1			DMI_RXn2			VSS			PCIE_RC_Rxn1
56		VSS			RSVD 54		PCIE_EP_RXp2				VSS		PCIE_EP_RXp0		VSS		DMI_RXn0				DMI_RXp2		DMI_RXp3	DMI_CLK100N		PCIE_RC_Rxn0		
57				RSVD 55		VSS			VSS			VSS				VSS			VSS			VSS			VSS			VSS

Table 35-5. Bottom View Right (Sheet 1 of 4)

	AK	AL	AM	AN	AR	AT	AU	AV	AW	AY	BA	BB	BC	BD	BE	BF	BG	BH	BJ	BK	BL	BM	BN	BP	BR	BT	BU
1			VSS			VSS			VSS			VSS				VSS			USB4_Dp			RSVD 66		RSVD 65			
2		SRDSI_2_P		SRDSI_3_P	RSVD 46		USB_RBIASp				USB0_Dn		USB1_Dp		USB3_Dn		USB4_Dn				MST_SMBCLK		RSVD 68			VSS	
3			SRDSI_2_N			RSVD 45			USB_CLK96P			USB1_Dn				USB3_Dp			VSS			VSS		RSVD 67			
4				SRDSI_3_N	VSS								USB2_Dn		USB2_Dp						MST_SMBDAT		OC1#/GPIO40		VSS		RSVD 59

Table 35-5. Bottom View Right (Sheet 2 of 4)

	AK	AL	AM	AN	AR	AT	AU	AV	AW	AY	BA	BB	BC	BD	BE	BF	BG	BH	BJ	BK	BL	BM	BN	BP	BR	BT	BU
5		SRDSI_1_N					USB_RBIASn		USB_CLK96N		USB0_Dp						VSS		VSS			RSVD 73		RSVD 60		RSVD 57	
6	VSS		VSS	VSS	VSS	VSS		VSS		VSS		VSS	VSS		VSS	VSS		VSS		VSS			VSS		GPIO27		RSVD 56
7																								GPIO24		VSS	
8		VCCAEP1P8AUX		VCC1P05		VCCSUS3P3_USB		VCCSUS_USB			VCCSUS_USB		VCCA1P8_USB			USB5_Dn		RSVD 74		VSS		VSS					
9		VSS		VCC1P05		VSSA_USB		VCCSUS3P3_USB			VCCA1P8_USB		VSS			USB5_Dp		TS1_IREF_N					VSS		GPIO56		GPIO47
10																			GPIO15	OC2# / GPIO41		GPIO30					
11		VCCAEPAUX		VCCA1P8_TS1		VCCASUS1P8_USB		VCCSUS3P3_USB			VSS		VCCA_USB			VSS							GPIO57			VSS	
12		VCCAEPAUX		VCCSUS1P8_USB		VSS		VCCSUS3P3_USB			VCCSUS3P3_USB		VCCA_USB			VSS	PLTRST#		VSS	OC0# / GPIO59		VSS			GPIO60		SML1ALERT# / GPIO74
13																						SUS_STAT# / GPIO61		SUS_CLK / GPIO62		GPIO72	
14		VSS		VSS		VCCSUS1P8_USB		VCCSUS1P8			VSS		VCCSUS3P3														
15		VCCAEPAUX		VCCA1P5_TS1		VCCSUS1P8_USB		VSS			VCCSUS3P3		VCCSUS3P3	VSS		GPIO12	MST_SMBALERT# / GPIO1		RSVD 78	GPIO31		VSS		SYS_PWROK		VSS	
16																						VSS			MEPWROK		GPIO8
17		VSS		VCCA1P5_TS0		VCCSUS1P8_USB		VCCSUS1P8			VCCSUS1P8		VSS	VSS		VCCSUS3P3	GPIO28		VSS	RI#			SLP_S4#			DRAMPWRGD	
18		VCCAEPAUX		VSS		VSS		VSS														OC3# / GPIO42					
19																							SLP_S3#		VSS		WAKE#
20		VCCEPAUX		VSS	VSS		VSS	VCCSUS		VSS	VSS		VCCSUS	VCCSUS		SLP_S5# / GPIO63	PWRBTN#		SML1CLK / GPIO58	SML1DAT / GPIO75		GPIO46					
21																							GPIO9			GPIO45	
22		VSS		VSS	VCCSUS		VCCSUS	VSS		VSS	VCCSUS		VSS	VCCSUS		VSS	GPIO10		VSS	GPIO43		VSS			GPIO73		VSS

Table 35-5. Bottom View Right (Sheet 3 of 4)

	AK	AL	AM	AN	AR	AT	AU	AV	AW	AY	BA	BB	BC	BD	BE	BF	BG	BH	BJ	BK	BL	BM	BN	BP	BR	BT	BU
23																						VSS		GPIO14		GPIO44	
24		VSS		VCC	VSS		VSS	VCCSUS																			
25										VSS	VCC3P3		VCC3P3	VSS		VCCSUS3P3_RTC	RSVD39		GPIO13	GPIO25		INTRUDER#		IVCC_RTC		RSVD53	
26		VCCEP		VSS	VCC		VCC	VCC3P3														GPIO26			RTCX1		RTCX2
27										VCC3P3	VSS		VCC1P8	VSS		VCC3P3_RTC	VSS		RTCRST#	PWROK			RSMRST#			VSS	
28		VSS		VCC	VSS		VSS	VSS														VSS					
29																							VSS		JTCK		JTDO
30		VCCEP		VSS	VCC		VCC	VCC3P3														JTDI					
31										VSS	VCC3P3		VSS	VCC1P8		VSS	SIU0_DCD#		SIU0_RXD	SRTCRST#			JTMS			RSVD38	
32		VSS		VCC	VCC		VSS	VSS														VSS			VSS		SIU0_DSR#
33										VCC3P3	VSS		VCC1P8	VSS		LFRAME#	LDRQ1# / GPIO23		LAD1	SIU0_RI#		SIU0_RTS#		SIU0_CTS#		RSVD77	
34		VCCEP		VCCA1P8_TS0	VCC		VCC	VCC3P3																			
35																						VSS		SIU1_DSR#		SIU1_CTS#	
36		VSS		VSS	VCC		VSS	VCC3P3		VSS	VCC3P3		VCC1P8	VSS		GPIO52	VSS		LDRQ0#	SIU1_DCD#		SIU1_RXD			VSS		SIU1_RTS#
37																							SIU1_RI#			WDT_TOUT#	
38		VCCA_DMI		VCCA_DMI	VCCA_DMI		VSS	VCC3P3		VSS	VCC3P3		VSS	SATA_LED#		GPIO17	GPIO32		PCICLK	VSS		BBS1 / GPIO51					
39																							UART_CLK		LAD0		LAD3
40		VSS		VSS		VCCA_DMI		VSS														VSS					

Table 35-5. Bottom View Right (Sheet 4 of 4)

	AK	AL	AM	AN	AR	AT	AU	AV	AW	AY	BA	BB	BC	BD	BE	BF	BG	BH	BJ	BK	BL	BM	BN	BP	BR	BT	BU
41		VCCA_DMI		VCCA_DMI		VSS		VCCA_SATA			VSS		VCCPCPU	SPI_CLK		VSS	VSS		BBS0	SIU1_TXD			GPIO33			LAD2	
42																						GPIO53			GPIO3		VSS
43		VSS		VSS		VCCA_DMI		VSS			VCCA_SATA		VCCA_SATA	VSS		GPIO19	GPIO50		GPIO4	GPIO2		VSS		SIU0_DTR#		SIU0_TXD	
44		VCCA_DMI		VCCA_DMI		SATA_ICOMPO		VCCA_SATA			RSVD 44		VSS														
45																						GPIO5		GPIO55		BMBUSY# / GPIO0	
46		VSS		PCIE_RC_TXp2		VSS		VCCA1P8_SATA			RSVD 43		SATA5_GP / TEMP_ALERT# /			GPIO35	VSS		RSVD 75	RSVD 76		VSS			GPIO54		VSS
47		PCIE_RC_TXp1		PCIE_RC_TXn2		SATA_VREF1P8		SATA4_TXP			VSS		SATA4_GP / GPIO16			RCIN#							GPIO1			GPIO7	
48																			SYS_RESET#	VSS		VSS					
49		PCIE_RC_TXn1		VSS		VSS		SATA4_TXN			SATA5_TXP		RSVD 47			GPIO21		VSS					GPIO34		SCLOCK / GPIO22		SLOAD / GPIO38
50		VSS		PCIE_RC_TXp3		PCIE_RC_TXn3		VSS			SATA5_TXN		GPIO20			VSS		SPI_MISO		INIT3_3V#		VSS					
51																								VSS		SDATAOUT1 / GPIO48	
52	VSS		VSS	VSS	VSS	VSS		VSS		VSS		VSS	VSS		VSS	THRMTRIP#		VSS		VSS			SIU1_DTR#		SDATAOUT0 / GPIO39		VSS
53		VSS					SATA5_RXN		SATA_CLK100N		PMSYNC						ADR/GPIO37		REF_CLK14			NRBOOTS		GPIO6		VSS	
54				PCIE_RC_RXn3	VSS								SPI_MOSI		RSVD 49						A20GATE		GPIO36		VSS		VSS
55			PCIE_RC_RXn2			SATA5_RXP			SATA_CLK100P			PECI				RSVD 50			SPI_CS0#			VSS		VSS			
56		PCIE_RC_RXp2		PCIE_RC_RXp3	SATA4_RXP		VSS				VSS		SPI_CS1#		RSVD 48		GPIO18				SERIRQ		VSS			VSS	
57			VSS			SATA4_RXN			VSS			CPUPWRGD				VSS			VSS			VSS		VSS			

Table 35-6. Top View Left (Sheet 1 of 4)

	BU	BT	BR	BP	BN	BM	BL	BK	BJ	BH	BG	BF	BE	BD	BC	BB	BA	AY	AW	AV	AU	AT	AR	AN	AM	AL	AK	AJ
1				RSVD 65		RSVD 66			USB4_Dp			VSS				VSS			VSS			VSS			VSS			VSS
2		VSS			RSVD 68		MST_SMBCLK				USB4_Dn		USB3_Dn		USB1_Dp		USB0_Dn				USB_RBIASp		RSVD 46	SRDSI_3_P		SRDSI_2_P		
3				RSVD 67		VSS			VSS			USB3_Dp				USB1_Dn			USB_CLK96P			RSVD 45			SRDSI_2_N			SRDSI_1_P
4	RSVD 59		VSS		OC1# / GPIO40		MST_SMBDAT						USB2_Dp		USB2_Dn								VSS	SRDSI_3_N				
5		RSVD 57		RSVD 60		RSVD 73			VSS		VSS						USB0_Dp		USB_CLK96N		USB_RBIASn					SRDSI_1_N		SRDSI_0_N
6	RSVD 56		GPIO27		VSS			VSS		VSS		VSS	VSS		VSS	VSS		VSS		VSS		VSS	VSS	VSS	VSS		VSS	
7		VSS		GPIO24																								
8						VSS		VSS		RSVD 74		USB5_Dn			VCCA1P8_USB		VCCSUS_USB			VCCSUS_USB		VCCSUS3P3_USB		VCC1P05		VCCAEP1P8AUX		
9	GPIO47		GPIO56		VSS					TS1_IREF_N		USB5_Dp			VSS		VCCA1P8_USB			VCCSUS3P3_USB		VSSA_USB		VCC1P05		VSS		
10						GPIO30		OC2# / GPIO41	GPIO15																			
11		VSS			GPIO57							VSS			VCCA_USB		VSS			VCCSUS3P3_USB		VCCASUS1P8_USB		VCCA1P8_TS1		VCCAEPAUX		
12	SML1ALERT# / GPIO74		GPIO60			VSS		OC0# / GPIO59	VSS		PLTRST#	VSS			VCCA_USB		VCCSUS3P3_USB			VCCSUS3P3_USB		VSS		VCCSUS1P8_USB		VCCAEPAUX		
13		GPIO72		SUS_CLK / GPIO62		SUS_STAT# / GPIO61																						
14															VCCSUS3P3		VSS			VCCSUS1P8		VCCSUS1P8_USB		VSS		VSS		
15		VSS		SYS_PWROK		VSS		GPIO31	RSVD 78		MST_SMBALERT# / GPIO11	GPIO12		VSS	VCCSUS3P3		VCCSUS3P3			VSS		VCCSUS1P8_USB		VCCA1P5_TS1		VCCAEPAUX		
16	GPIO8		MEPWROK			VSS																						
17		DRAMPWRGD			SLP_S4#			RI#	VSS		GPIO28	VCCSUS3P3		VSS	VSS		VCCSUS1P8			VCCSUS1P8		VCCSUS1P8_USB		VCCA1P5_TS0		VSS		
18						OC3# / GPIO42														VSS		VSS		VSS		VCCAEPAUX		

Table 35-6. Top View Left (Sheet 2 of 4)

	BU	BT	BR	BP	BN	BM	BL	BK	BJ	BH	BG	BF	BE	BD	BC	BB	BA	AY	AW	AV	AU	AT	AR	AN	AM	AL	AK	AJ
19	WAKE#		VSS		SLP_S3#																							
20						GPIO46		SML1DAT / GPIO75	SML1CLK / GPIO58		PWRBTN#	SLP_S5# / GPIO63		VCCSUS	VCCSUS		VSS	VSS		VCCSUS	VSS		VSS	VSS		VCCEPAUX		VSS
21		GPIO45			GPIO9																							
22	VSS		GPIO73			VSS		GPIO43	VSS		GPIO10	VSS		VCCSUS	VSS		VCCSUS	VSS		VSS	VCCSUS		VCCSUS	VSS		VSS		VCCEPAUX
23		GPIO44		GPIO14		VSS																						
24																				VCCSUS	VSS		VSS	VCC		VSS		VCCEPAUX
25		RSVD53		IVCC_RTC		INTRUDER#		GPIO25	GPIO13		RSVD39	VCCSUS3P3_RTC		VSS	VCC3P3		VCC3P3	VSS										
26	RTCX2		RTCX1			GPIO26														VCC3P3	VCC		VCC	VSS		VCCEP		VSS
27		VSS			RSMRST#			PWROK	RTCRST#		VSS	VCC3P3_RTC		VSS	VCC1P8		VSS	VCC3P3										
28						VSS														VSS	VSS		VSS	VCC		VSS		VCCEP
29	JTDO		JTCK		VSS																							
30						JTDI														VCC3P3	VCC		VCC	VSS		VCCEP		VSS
31		RSVD38			JTMS			SRTCRST#	SIU0_RXD		SIU0_DCD#	VSS		VCC1P8	VSS		VCC3P3	VSS										
32	SIU0_DSR#		VSS			VSS														VSS	VSS		VCC	VCC		VSS		VCCEP
33		RSVD77		SIU0_CTS#		SIU0_RTS#		SIU0_RI#	LAD1		LDRQ1# / GPIO23	LFRAME#		VSS	VCC1P8		VSS	VCC3P3										
34																				VCC3P3	VCC		VCC	VCCA1P8_TS0		VCCEP		VSS
35		SIU1_CTS#		SIU1_DSR#		VSS																						
36	SIU1_RTS#		VSS			SIU1_RXD		SIU1_DCD#	LDRQ0#		VSS	GPIO52		VSS	VCC1P8		VCC3P3	VSS		VCC3P3	VSS		VCC	VSS		VSS		VCCEP

Table 35-6. Top View Left (Sheet 3 of 4)

	BU	BT	BR	BP	BN	BM	BL	BK	BJ	BH	BG	BF	BE	BD	BC	BB	BA	AY	AW	AV	AU	AT	AR	AN	AM	AL	AK	AJ
37		WDT_TOUT#			SIU1_RI#																							
38						BBS1 / GPIO51		VSS	PCICLK		GPIO32	GPIO17		SATA_LED#	VSS		VCC3P3	VSS		VCC3P3	VSS		VCCA_DMI	VCCA_DMI		VCCA_DMI		VSS
39	LAD3		LAD0		UART_CLK																							
40						VSS														VSS		VCCA_DMI		VSS		VSS		
41		LAD2			GPIO33			SIU1_TXD	BBS0		VSS	VSS		SPI_CLK	VCCP CPU		VSS			VCCA_SATA		VSS		VCCA_DMI		VCCA_DMI		
42	VSS		GPIO3			GPIO53																						
43		SIU0_TXD		SIU0_DTR#		VSS		GPIO2	GPIO4		GPIO50	GPIO19		VSS	VCCA_SATA		VCCA_SATA			VSS		VCCA_DMI		VSS		VSS		
44															VSS		RSVD 44			VCCA_SATA		SATA_ICOMPO		VCCA_DMI		VCCA_DMI		
45		BMBUSY# / GPIO0		GPIO55		GPIO5																						
46	VSS		GPIO54			VSS		RSVD 76	RSVD 75		VSS	GPIO35			SATA5_GP / TEMP_ALERT# /		RSVD 43			VCCA 1P8_SATA		VSS		PCIE_RC_TXp2		VSS		
47		GPIO7			GPIO1							RCIN#			SATA4_GP / GPIO16		VSS			SATA4_TXP		SATA_VREF1P8		PCIE_RC_TXn2		PCIE_RC_TXp1		
48						VSS		VSS	SYS_RESET#																			
49	SLOAD / GPIO38		SCLOCK / GPIO22		GPIO34					VSS		GPIO21			RSVD 47		SATA5_TXP			SATA4_TXN		VSS		VSS		PCIE_RC_TXn1		
50						VSS		INIT3_3V#		SPI_MISO		VSS			GPIO20		SATA5_TXN			VSS		PCIE_RC_TXn3		PCIE_RC_TXp3		VSS		
51		SDATAOUT1 / GPIO48		VSS																								
52	VSS		SDATAOUT0 / GPIO39		SIU1_DTR#			VSS		VSS		THRMTRIP#	VSS		VSS	VSS		VSS		VSS		VSS	VSS	VSS	VSS		VSS	
53		VSS		GPIO6		NRBOOTS			REF_CLK14		ADR/ GPIO37						PMSYNC		SATA_CLK100N		SATA5_RXN					VSS		PCIE_RC_RXp1
54	VSS		VSS		GPIO36		A20GATE						RSVD 49		SPI_MOSI								VSS	PCIE_RC_RXn3				

Table 35-6. Top View Left (Sheet 4 of 4)

	BU	BT	BR	BP	BN	BM	BL	BK	BJ	BH	BG	BF	BE	BD	BC	BB	BA	AY	AW	AV	AU	AT	AR	AN	AM	AL	AK	AJ
55				VSS		VSS			SPI_CS0#			RSVD 50				PECI			SATA_CLK100P			SATA5_RXP			PCIE_RC_RXn2			PCIE_RC_Rxn1
56		VSS			VSS		SERIRQ				GPIO18		RSVD 48		SPI_CS1#		VSS						SATA4_RXP	PCIE_RC_RXp3		PCIE_RC_RXp2		
57				VSS		VSS			VSS			VSS				CPUPWRGD			VSS			SATA4_RXN			VSS			VSS

Table 35-7. Top View Right (Sheet 1 of 4)

	AH	AG	AF	AE	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
1			GBE_CLK100P			VSS			VSS			VSS				VSS			RSVD 83			RSVD 3		RSVD 4			
2		VSS		VSS	SRDSO_2_P		SRDSO_1_P				SFP0_I2C_DATA		GBE1_SWDP0		SFP1_I2C_CLK		SFP3_I2C_CLK				VSS		RSVD 0			RSVD 2	
3			GBE_CLK100N			SRDSO_1_N			SRDSO_0_P			SFP0_I2C_CLK				SRDS0_SD			GBE1_SWDP1			SRDS2_SD		VSS			
4				VSS	SRDSO_2_N								GBE3_SWDP0		SFP3_I2C_DATA						GBE2_SWDP1		GBE2_LED		RSVD 6		VSS
5		SRDSI_0_P					SRDSO_0_N		VSS		VSS						EP_MAIN_PWR_OK		VSS			RSVD 85		GBE_EE_DI		RSVD 1	
6	VSS		VSS	VSS	VSS	VSS		VSS		VSS		VSS	GBE2_SWDP0		GBE0_LED	VSS		VSS		RSVD 84			VSS		SRDS1_SD		RSVD 8
7																								VSS		GBE_AUX_PWR_AVAIL	
8		GBE_IRCOMP		RSVD 63		RSVD 64		SFP1_I2C_DATA			SFP2_I2C_DATA		SFP2_I2C_CLK			GBE0_SWDP1		GBE_AUX_PWR_OK		RSVD 86		GBE_EE_DO					
9		VCCAEP1P8AUX		VSS		VSS		GBE0_SWDP0			GBE3_LED		VSS			SRDS3_SD		VSS					RSVD 82		RSVD 79		VSS
10																			GBE_EE_SK	GBE3_SWDP1		PCIE_EP_RST#					
11		VCCAEP1P8AUX		GBE_VREF1P8		SRDSO_3_P		VSS			GBE_SMBDAT		GBE1_LED			VSS							RSVD 80			RSVD 81	
12		VSS		VSS		SRDSO_3_N		GBE_SMBCLK			GBE_SMBALRT#		GBE_EE_CS#			EP_SMBDAT	EP_CRU_EN		VSS	GBE_WAKE#		VSS			EP_SMBALRT#		EP_JTCK
13																						RSVD 16		EP_JTRST#		EP_SMBCLK	

Table 35-7. Top View Right (Sheet 2 of 4)

	AH	AG	AF	AE	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
14		VCCAEPAUX		VCCPEP3P3AUX		VSS		RSVD 87			VSS		RSVD 88															
15		VSS		VSS		VCCPEP3P3AUX		EP_JTDI			RSVD 37		RSVD 36		VSS	EP_JTDO	RSVD 90		RSVD 89	VSS		VSS		RSVD 22		RSVD 5	
16																						VSS			RSVD 13		VSS
17		VCCPEP3P3AUX		VCCPEP3P3AUX		VCCEPAUX		VSS			VCCAEPAUX		VSS	RSVD 12		RSVD 10	VSS		EP_JTMS	RSVD 7			RSVD 26			RSVD 11	
18		VSS		VSS		VSS		VSS														RSVD 9					
19																							VSS		RSVD 28		VSS
20		VCCEPAUX		VSS	VCCEPAUX		VSS	VSS		VSS	VCCAEPAUX		RSVD 20	VSS		RSVD 15	RSVD 24		RSVD 14	VSS		RSVD 30					
21																							RSVD 34			RSVD 21	
22		VSS		VCCEPAUX	VSS		VCCEPAUX	VCCEPAUX		VCCEPAUX	VCCAEP1P8_CRU		VCCAEP1P8_CRU	VCCAEPAUX		RSVD 33	VSS		CRU_CLK100N	RSVD 17		RSVD 18			VSS		RSVD 27
23																						RSVD 19		RSVD 23		RSVD 25	
24		VCCEPAUX		VCCEPAUX	VCCEP1P0_CRU		VSS	VSS																			
25										VSS	VCCEPAUX		VSS	RSVD 71		RSVD 70	RSVD 69		CRU_CLK100P	RSVD 35		VSS		RSVD 31		RSVD 29	
26		VCCEP		VSS	VCCEP1P0_CRU		VSS	VSS														VSS			VSS		VSS
27										VCCEP	VSS		VCCAEPAUX	VSS		VSS	RSVD 72		VSS	RSVD 32			PCIE_EP_RXn15			PCIE_EP_RXp15	
28		VSS		VCCEP	VSS		VCCEP	VCCEP														VSS					
29																							PCIE_EP_RXn14		PCIE_EP_RXp14		VSS
30		VCCEP		VSS	VCCEP		VCCEP	VCCEP														VSS					
31										VSS	VCCEP		VSS	PCIE_EP_TXp15		PCIE_EP_TXn15	VSS		PCIE_EP_TXp14	PCIE_EP_TXn14			PCIE_EP_RXn13			PCIE_EP_RXp13	

Table 35-7. Top View Right (Sheet 3 of 4)

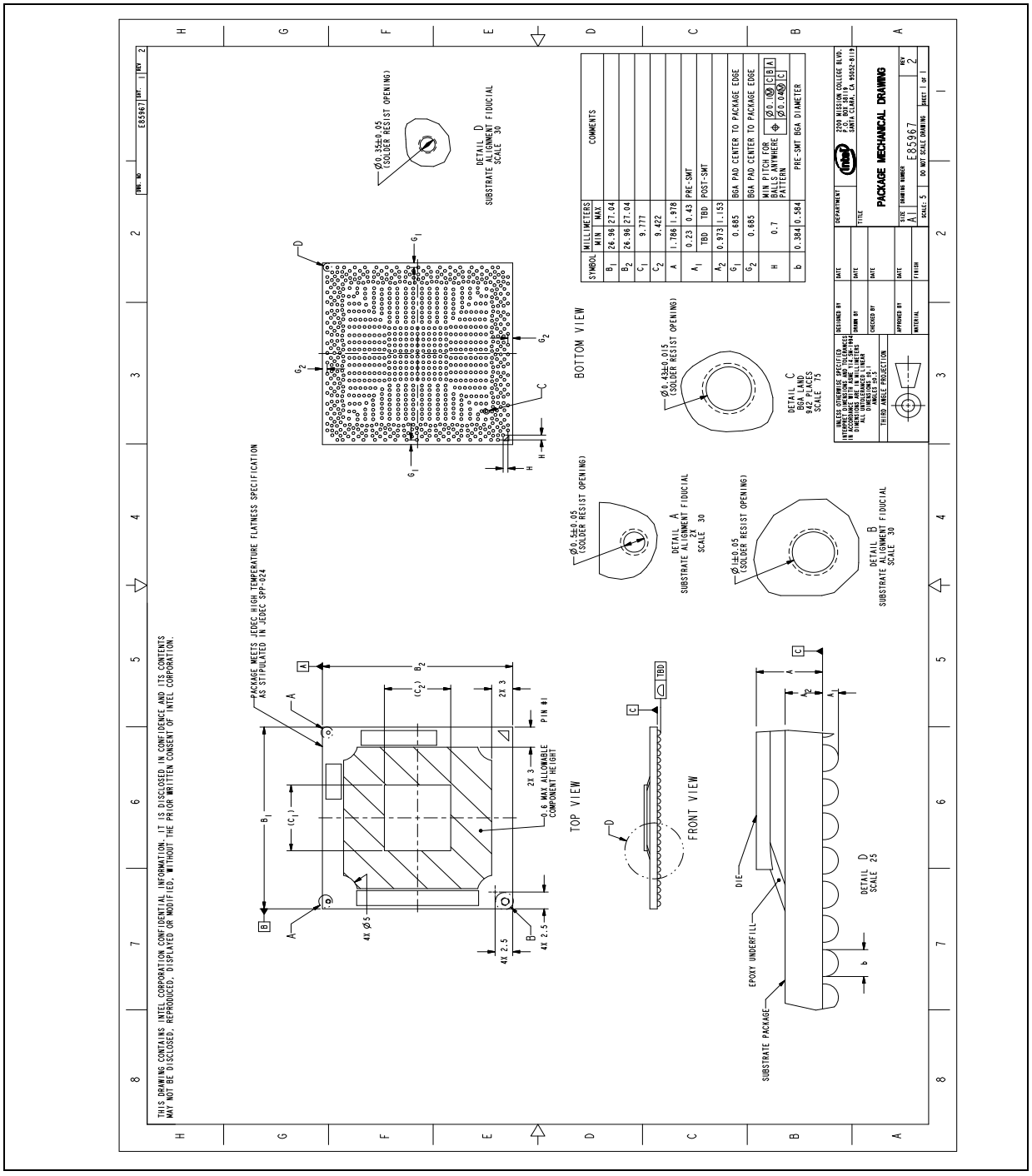
	AH	AG	AF	AE	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
32		VSS		VCCEP	VSS		VSS	VSS														VSS			VSS		VSS
33										VCCEP	VSS		PCIE_EP_TXp13	PCIE_EP_TXn13		VSS	PCIE_EP_TXn12		PCIE_EP_TXp12	VSS		VSS		PCIE_EP_RXn12		PCIE_EP_RXp12	
34		VCCEP		VSS	VCCEP		VCCEP	VCCEP																				
35																						VSS		PCIE_EP_RXp11		PCIE_EP_RXn11	
36		VSS		VCCAEP_PE	VSS		VSS	VSS		VSS	VCCEP		VSS	PCIE_EP_TXp11		PCIE_EP_TXn11	VSS		PCIE_EP_TXp10	PCIE_EP_TXn10		VSS			PCIE_EP_RXp10		VSS
37																							PCIE_EP_RXp9			PCIE_EP_RXn10	
38		VCCAEP_PE		VSS	VCCAEP_PE		VCCAEP_PE	VCCAEP_PE		VSS	PCIE_EP_ICOMPI		VSS	PCIE_EP_TXn8		PCIE_EP_TXp8	VSS		PCIE_EP_TXp9	PCIE_EP_TXn9		VSS					
39																							PCIE_EP_RXp8		PCIE_EP_RXn9		VSS
40		VSS		VCCAEP_PE		VSS		VSS														VSS					
41		VCCAEP_PE		VSS		VCCAEP_PE		VCCAEP_PE			VSS		PCIE_EP_CLK100P	VSS		RSVD61	VSS		PCIE_EP_TXn7	PCIE_EP_TXp7			PCIE_EP_RXn8			PCIE_EP_RXp7	
42																						VSS			PCIE_EP_RXn7		VSS
43		VCCAEP_PE		VCCAEP_PE		VCCAEP_PE		VCCAEP1P8_PE			PCIE_EP_CLK100N		VSS	RSVD62		VSS	PCIE_EP_TXn6		PCIE_EP_TXp6	VSS		VSS		PCIE_EP_RXn6		PCIE_EP_RXp6	
44		VCCA1P8_DMI		VSS		VSS		PCIE_EP_VREF1P8			VSS		VSS														
45																						VSS		PCIE_EP_RXp5		PCIE_EP_RXn5	
46		DMI_VREF		RSVD41		RSVD42		VSS			VSS		PCIE_EP_TXn1			PCIE_EP_TXp1		VSS	PCIE_EP_TXn5	PCIE_EP_TXp5		VSS			PCIE_EP_RXp4		VSS
47		VSS		DMI_IRCOMP		VSS		DMI_TXn1			DMI_TXn0		VSS											VSS		PCIE_EP_RXn4	
48																PCIE_EP_TXp2			VSS	PCIE_EP_TXp4		VSS					
49		PCIE_RC_TXp0		VSS		DMI_TXp2		DMI_TXp1			DMI_TXp0		PCIE_EP_TXn0			PCIE_EP_TXn2		PCIE_EP_TXp3					VSS		VSS		VSS

Table 35-7. Top View Right (Sheet 4 of 4)

	AH	AG	AF	AE	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
50		PCIE_RC_TXn0		DMI_TXp3		DMI_TXn2		VSS			VSS		PCIE_EP_TXp0			VSS		PCIE_EP_TXn3		PCIE_EP_TXn4		VSS					
51																								PCIE_EP_RXp3		PCIE_EP_RXn3	
52	VSS		DMI_TXn3	VSS	VSS	VSS		VSS		VSS		VSS	VSS		VSS	VSS		VSS		VSS			VSS		VSS		VSS
53		PCIE_RC_RXp0					VSS		DMI_RXp1		VSS						PCIE_EP_RXp1		VSS			VSS		TS0_IREF_N		VSS	
54				DMI_CLK100P	DMI_RXn3								VSS		PCIE_EP_RXn0						PCIE_EP_RXn2		VSS		RSVD 52		VSS
55			VSS			DMI_RXn2			DMI_RXn1			DMI_RXp0				PCIE_EP_RXn1			VSS			VSS		RSVD 51			
56		PCIE_RCRxn0		DMI_CLK100N	DMI_RXp3		DMI_RXp2				DMI_RXn0		VSS		PCIE_EP_RXp0		VSS				PCIE_EP_RXp2		RSVD 54			VSS	
57			VSS			VSS			VSS			VSS				VSS			VSS			VSS		RSVD 55			

35.3 Package Mechanical Information

Figure 35-1. Mechanical Package



§ §

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Chipsets ***category:***

Click to view products by Intel ***manufacturer:***

Other Similar products are found below :

AC5500 S LH3N NU3210MC S LJEF DH8920CC S LK5Y AC5520 S LH3P GLCM236 S R2CE DH8950CL S LKCK AC82G45 S LB84 GT24L16M1Y20 GT24L24A2Y GT24L16A2Y GT22L16A2Y GT24L24A2Y16 GT32L32M4W40 GT5DL24A2W GT5DL24A1Y80 AC82GL40 S LB95 BD82C604 S LJKJ BD82HM55 S LGZS BD82HM76 S LJ8E DH8900CC S LJW2 DH8903CC S LJVZ DH8920CC S LJVX GT30L32S4W GT21L16S2Y GT22L16M1Y GT31L16M1Y80 GT32L24A180 GT32L24F0210 GT30L24A3W GT24L24A2W16 GT60L16M2K4 GT21L24S1W GT30L24T3Y GT32L32S0140 GT31L16S2W80 GT22L16U1Y40 SD16S1Y GT20L16S1Y GT20L16P1Y GT20L24F6Y GT30L24M1Z GT32L32M0180 GT21L16S2W