Intel® Quark™ Microcontroller D2000

Datasheet

May 2017

Document Number: 333577-005EN

Contents

Figures

Tables

§

Revision History

Date	Revision	Description
May 2017	005	Minor corrections in Sections 17.3.1.3 and 17.3.1.12
March 2017	004	Corrected typos in tables in Section 4.4.1, "IO DC specifications"
November 2016	003	Updated to correct information & add in power consumption figures
January 2016	002	Initial public release
December 2015	001	Internal release available by NDA

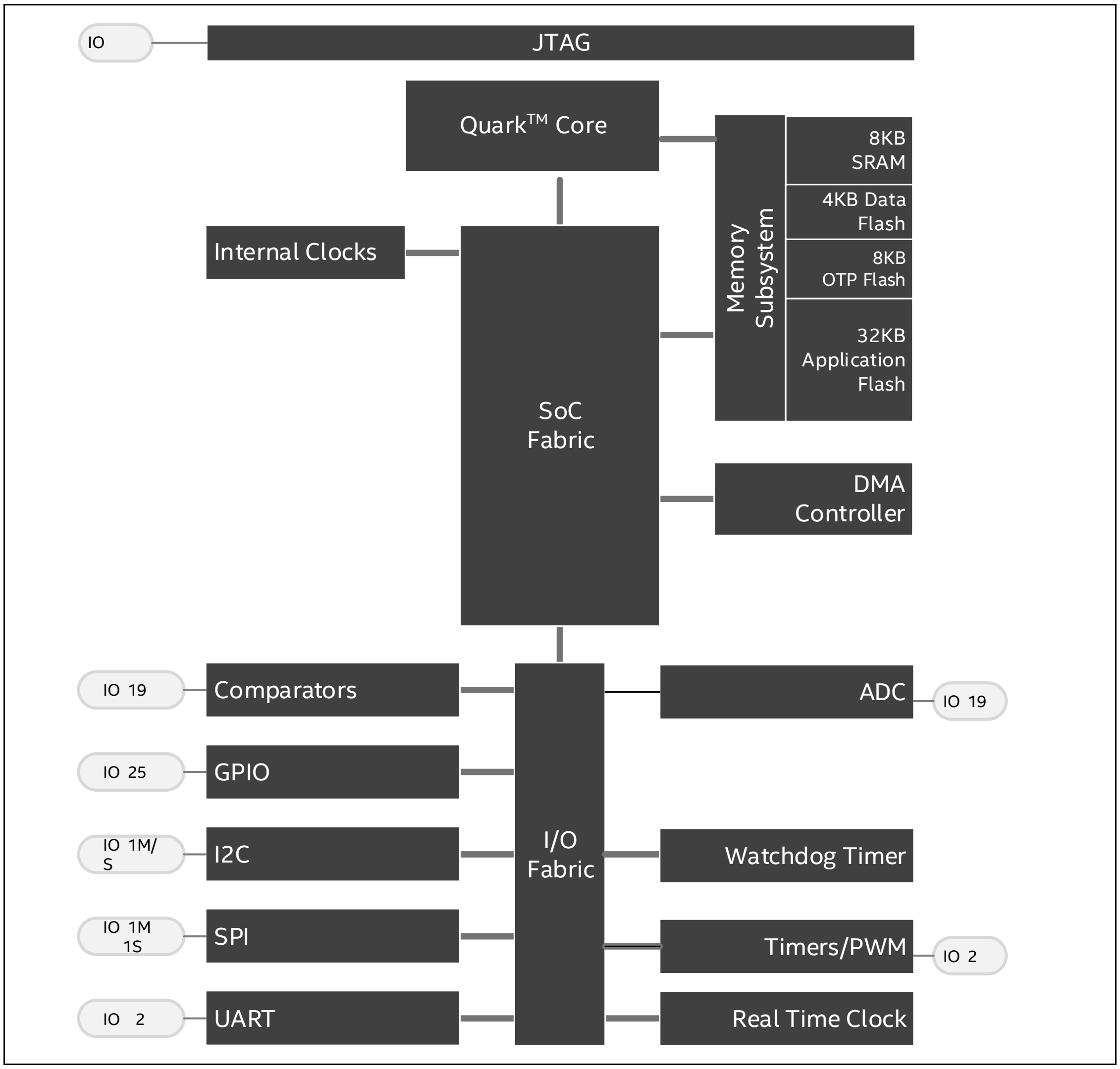
§

1 Introduction

The Intel® Quark™ microcontroller D2000 is an ultra-low power Intel Architecture (IA) system-on-chip (SoC) that integrates an Intel® Quark processor core, Memory Subsystem with on-die volatile and non-volatile storage, and I/O interfaces into a single low-cost SoC solution.

Figure 1 shows the system level block diagram of the SoC. Refer to the subsequent chapters for detailed information on the individual functional blocks.

Figure 1. SoC Block Diagram



1.1 Terminology

Table 1. Table of Acronyms

Acronym	Full Name
ACG	Autonomous Clock Gating
ADC	Analog to Digital Converter
AHB	High-performance Bus
AI	Analog Input
AON	Always On
APB	Advanced Peripheral Bus
APIC	Advanced Programmable Interrupt Controller
CCR	Current Count Register
CCU	Clock Control Unit
DCG	Dynamic Clock Gating
DCR	Direct Current Resistance
DFS	Dynamic Frequency Scaling
DLAB	Divisor Latch Access Bit
DLL	Divisor Latch Low
DMA	Direct Memory Access
DTCM	Data Tightly-Coupled Memory
EOI	End of Interrupt
ESR	Equivalent Series Resistance
FIFO	First In First Out
FPR	Flash Protection Region
FPU	Floating Point Unit
FSB	Front-side bus
FST	Forced swim test
FW	Firmware
GDT	Global Descriptor Table Register
GND	Ground
GPIO	General-purpose input/output
I^2C	Inter-Integrated Circuit
ICR	Input Capture Register
IDT	Interrupt Descriptor Table Register
IMR	Isolated Memory Region
IRQ	Interrupt Input

Acronym	Full Name
IRR	Interrupt Request Register
ISR	In-Service Register
ITCM	Instruction TCM
JTAG	Joint Test Action Group
MMIO	Memory Mapped IO
MSI	Message Signaled Interrupts
MVIC	Intel® Quark™ microcontroller D2000 Interrupt Controller
NVM	System Non-Volatile-Memory
OTP	One Time Programmable
PCB	Printec Circuit Board
PMU	Power Management Unit
PPR	Processor Priority Register
PVT	Process, Voltage and Temperature
PWM	Pulse Width Modulation
QFN	Quad Flat No-Leads
RBR	Receive Buffer Registers
RTC	Real Time Clock
SCL	Serial Clock
SCM	System Control Module
SCSS	System Control Subsystem
SDA	Sarial Data Signal
SIVR	Spurious Interrupt Vector Register
SoC	System on Chip
SPI	Serial Peripheral Interface Bus
SRAM	Static Random Access Memory
SW	Software
TCK	Test Clock Signal
TCM	Tightly Coupled Memory
TDO	Test Data Out Signal
THR	Transmit Holding Register
TMS	Test Mode Select
UART	Universal asynchronous receiver/transmitter
VIL	Voltage Input Level
WDT	Watchdog Timer
XTAL	Crystal

1.2 Feature Overview

1.2.1 Clock Oscillators

- 32 MHz Clock (system clock) generated by on-die Hybrid Oscillator which works in either of the following:
  - Silicon mode (external crystal not needed) (generates 4/8/16/32 MHz clock output as configured)
  - Crystal mode (external 32MHz crystal required)
- 32.768 kHz RTC Clock generated by on-die RTC Crystal oscillator (external 32.768kHz crystal required). SoC is designed to work without RTC clock, if there is no use-case for RTC clock.

1.2.2 Quark Processor Core

- 32 MHz Clock Frequency
- 32-bit Address Bus
- Pentium 586 ISA Compatible without x87 Floating Point Unit
- Integrated Local APIC and I/O APIC
- 1 32-bit timer in Local APIC running with system/core clock

1.2.3 Memory Subsystem

- 32 KB of 64b wide on-die Flash
- Supports Page Erase and Program cycles
- Supports configurable wait states to allow Flash to run at various frequencies. At 32MHz, 2-wait-states are introduced for all accesses
- 4 configurable Protection regions for Flash access control
- 8 KB Code OTP with independent read-disable of the two 4KB regions
- 4 KB Data Flash memory
- 8 KB of on-die SRAM with 64b interface with 0-wait state in case of no arbitration conflict
- 4 configurable Protection regions for SRAM access control
- Unique part identifier – see note Memory section on its use & configuration

1.2.4 I^{2}C

- One I^{2}C Interface
- Three I^{2}C speeds supported : Standard Mode (100 Kbps), Fast Mode (400 Kbps) and Fast Mode Plus (1 Mbps)
- 7-bit and 10-bit Addressing Modes Supported
- Supports Master or Slave operation
- FIFO mode support (16B TX and RX FIFO's)

- Supports HW DMA with configurable FIFO thresholds

1.2.5 UART

- Two 16550 compliant UART interfaces
- Supports baud rates from 300 to 2M with less than 2% frequency error
- Support for hardware and software flow control
- FIFO mode support (16B TX and RX FIFO's)
- Supports HW DMA with configurable FIFO thresholds
- Supports 9-bit serial operation mode
- Supports RS485
- Support for DTR/DCD/DSR/RI Modem Control Pins through GPIO pins controlled by Software

1.2.6 SPI

- One SPI Master Interfaces with support for SPI clock frequencies up to 16 MHz
- One SPI Slave Interface with support for SPI clock frequencies up to 3.2 MHz
- Support for 4-bit up to 32-bit Frame Size
- Up to four Slave Select pins per Master interface
- FIFO mode support (Independent 32B TX and RX FIFO's)
- Supports HW DMA with configurable FIFO thresholds

1.2.7 DMA Controller

- Provides 2 Unidirectional Channels
- Provides support for 16 HW Handshake Interfaces
  - TX and RX channels of I^{2}C controller, SPI Slave controller, SPI Master controller, two UART controllers use this interface
- Supports Memory to Memory, Peripheral to Memory, Memory to Peripheral and Peripheral to Peripheral transfers
- Dedicated Hardware Handshaking interfaces with peripherals plus Software Handshaking Support
- Supports Single and Multi-Block Transfers

1.2.8 GPIO Controller

- Provides 25 independently configurable GPIOs
- All GPIOs are interrupt capable supporting level sensitive and edge triggered modes
- Debounce logic for interrupt source
- All 25 GPIOs are Always-on interrupt and wake capable

1.2.9 Timers

- Two 32-bit Timers running at system clock (running in timer mode or PWM mode)
- Supports an additional 32-bit Always-On Counter running with 32.768 kHz clock
- Supports an additional 32-bit Always-On Periodic Timer running with 32.768 kHz clock and with interrupt and wake capability

1.2.10 Pulse Width Modulation (PWM)

- Two 32-bit Timers running at system clock can be configured to generate two PWM outputs

1.2.11 Watchdog Timer

- Configurable Watchdog timer with support to trigger an interrupt and/or a system reset upon timeout

1.2.12 Real Time Clock (RTC)

- 32-bit Counter running from 1Hz up to 32.768 kHz
- Supports interrupt and wake event generation upon match of programmed value
- Only requires 32.768 kHz clock to be running to generate interrupt and wake events

1.2.13 Analog to Digital Convertor (ADC)

- 19 Analog Input channels
- Selectable 6/8/10/12-bit resolution
- Supports maximum of 2.28 Mega Samples Per Second (MSps) at 12-bit resolution and 4 MSps at 6-bit resolution)
- Differential Non-Linearity (DNL) of +/- 1.0 LSB
- Integral Non-Linearity (INL) of +/- 2.0 LSB
- SINAD of 68 dBFS
- Offset Error of +/- 2 LSB (calibration enabled), +/- 64 LSB (calibration disabled)
- Full-scale input range of 0 to AVDD.

1.2.14 Analog Comparators

- Provides 19 Analog Comparators
- Six high performance comparators
- 13 low power comparators
- Configurable polarity

- Interrupt and Wake Event capable

1.2.15 Interrupt Routing

- Configurable Routing of SoC Interrupts with capability to route to the Interrupt Controller of the Quark Processor.
- SoC events can be routed as: Interrupts to the Quark Processor, debug break events to the Quark Processor or SoC Warm Reset requests.

1.2.16 Power Management

- SoC System States : RUN, Low Power Compute, HALT, Low Power Wait, Deep Sleep (RTC or NORTC) state.
- Processor States : C0 – C2
- Supports Coin-cell Battery source (2.0V to 3.6V range)

Scenario	Max Current Draw
Active/RUN state: All SoC components, including ADC, Comparators, clock oscillators, and peripherals are enabled and core runs at 32MHz.	< 30 mA
Idle/Sleep state: Most SoC components, such as ADC, hybrid oscillator, RTC oscillator, and peripherals are powered down or clock-gated. Core is halted. Only one low-power comparator is enabled for wake.	< 3.5 µA

1.2.17 Package

40-pin Quad Flat No-Leads (QFN) package.

§

2 Physical Interfaces

2.1 Pin States Through Reset

All functional IOs will come up in input mode after reset except JTAG TDO output which is kept tristated.

All Digital IO include a configurable pullup (49 KOhm typ; 34K-74 KOhm range) with pull-up disabled by default, except for F_20, F_22, F_23 pins (TRST_N, TMS, TDI)) where pull-up is enabled by default.

The state of all IOs is retained whenever SoC goes into low power states.

2.2 External Interface Signals

The following table gives the definition of external interface signals of Intel® Quark™ microcontroller D2000. Not all interfaces are available simultaneously through external pins of Intel® Quark™ microcontroller D2000. For pin multiplexing options, refer to Chapter 3.

Table 2. List of User Mode External Interfaces

Interface	Pin Name	Type	Description
Power	VSS	Ground	QFN package Exposed Tab
	PVDD	Supply	2.0-3.6 V unregulated battery supply rail input (can lower to 1.8V if analog comparators are not used). This rail is used only by internal voltage regulator. There is a mechanism to disable internal voltage regulator and feed IOVDD/AVDD/DVDD by platform directly. PVDD is to be supplied even if Internal voltage regulator is not enabled as internal VR is used to generate internal voltage reference for comparators.
	AVDD	Supply	2.0-3.6V Analog Voltage Rail Input powering both ADC and Comparator - ADC supports 1.8V to 3.3V range, but Comparator only supports 2V to 3.3V. AVDD can be an AC isolated version of PVDD.
	IOVDD	Supply	Analog (Driver) side Voltage Rail Input for IO ring (1.8V to 3.3V Nominal +/- 10%). All digital IO pads use IOVDD only. IOVDD can be an AC-isolated version of PVDD but isolation is not mandatory.
	VSENSE	Analog input	Voltage Regulator Voltage Sense Input - Feedback of Load side of inductor.
	GSENSE	Analog input	Core power ground sense

Interface	Pin Name	Type	Description
	LX	Supply	Core voltage regulator output
	VREN	Analog input	Internal Voltage regulator enable: PVDD = enable VSS/GND = disable VREN cannot be dynamically changed. VREN has to be stable/static when PVDD becomes stable.
	DVDD, DVDD_2	Supply	1.8 V (nominal) +/- 10% regulated core power supply. In Deep Sleep state, it can be configured to drop to 1.35V (nominal) +/- 10% to reduce sleep/leakage power.
Clocking	HYB_XTALI	Logic input	Crystal/oscillator input for System Clock. If no external XTAL is connected, this pin should be grounded.
	HYB_XTALO	Logic output	Crystal output for System clock. If no external XTAL is connected, keep these pins as no-connect.
	RTC_XTALI	Logic input	32.768 kHz Crystal/oscillator input for RTC clock. If RTC XTAL is not connected, this pin has to be tied to VSS.
	RTC_XTALO	Logic output	Crystal output for RTC clock. If RTC XTAL is not connected, this pin should be left unconnected.
	SYS_CLK_OUT	Logic Output	Divided (1:1, 1:2, 1:4) version of 32MHz system clock output
	RTC_CLK_OUT	Logic Output	32.768 kHz RTC clock output
Reset	RST_N	Analog input	Active Low reset input with Hysteresis. RST_N has to be asserted low till input voltage rails (PVDD/AVDD/IOVDD/DVDD) reach specified operating range. RST_N is connected to an internal comparator which compares RST_N voltage level to an internal reference to assert/deassert internal SOC reset. There is an internal mechanism to assert reset if PVDD is power recycled (power on reset) irrespective of RST_N voltage level.
GPIO	GPIO[24:0]	Logic I/O	General purpose I/O
I^{2}C	I2C_SCL	Logic I/O	Open drain clock
	I2C_SDA	Logic I/O	Open drain data
PWM	PWM1	Logic Output	PWM Output 1
	PWM0	Logic Output	PWM Output 0
UART	UART_A_TXD	Logic output	UART A single-ended Transmit data (RS232 or RS485).

Interface	Pin Name	Type	Description
	UART_A_RXD	Logic input	UART A single-ended Receive data (RS232 or RS485).
	UART_A_RTS	Logic output	UART A Request To Send (RS232)
	UART_A_CTS	Logic input	UART A Clear To Send (RS232)
	UART_A_DE	Logic Output	UART A Driver Enable (RS485 mode). Used to control external RS-485 device. Polarity is configurable. This is multiplexed onto UART_A_RTS pin depending on RS485 or RS232 mode of operation.
	UART_A_RE	Logic Output	UART B Receiver Enable (RS485 mode). Used to control external RS-485 driver. Polarity is configurable. This is multiplexed onto UART_B_CTS pin depending on RS485 or RS232 mode of operation.
	UART_B_TXD	Logic output	UART B single-ended Transmit data (RS232 or RS485).
	UART_B_RXD	Logic input	UART B single-ended Receive data (RS232 or RS485).
	UART_B_RTS	Logic output	UART B Request To Send (RS232)
	UART_B_CTS	Logic input	UART B Clear To Send (RS232)
	UART_B_DE	Logic Output	UART B Driver Enable (RS485 mode). Used to control external RS-485 driver. This is multiplexed onto UART_B_RTS pin depending on RS485 or RS232 mode of operation.
	UART_B_RE	Logic Output	UART A Receiver Enable (RS485 mode). Used to control external RS-485 driver. This is multiplexed onto UART_B_CTS pin depending on RS485 or RS232 mode of operation.
Slave SPI	SPI_S_SCLK	Logic input	Slave SPI Clock
	SPI_S_SDIN	Logic input	Slave SPI Receive data
	SPI_S_SCS	Logic input	Slave SPI Slave Chip Select
	SPI_S_SDOUT	Logic output	Slave SPI Transmit data
Master SPI	SPI_M_SCLK	Logic output	Master SPI Clock
	SPI_M_TXD	Logic output	Master SPI Transmit data
	SPI_M_SS[3:0]	Logic output	Master SPI Slave Selects
	SPI_M_RXD	Logic input	Master SPI Receive data

Interface	Pin Name	Type	Description
Analog	AI[18:0]	Analog input	Comparator/ADC inputs. AI[18:0] are connected to both ADC and comparator inputs. AI[5:0] are connected to 6 fast response analog comparators. AI[18:6] are connected to 13 slow response low power comparators, Since fast response comparators are powered off in low power states, any analog signal that needs to be capable of waking SoC should be connected to one of AI[18:6].
	AR	Analog input	Comparator Reference Voltage Input (internal reference from RAR also available).
JTAG	TRST_N	Logic input	TAP controller reset. A pull-up is enabled by default.
	TDI	Logic input	TAP data input. A pull-up is enabled by default.
	TMS	Logic input	TAP mode select. A pull-up is enabled by default.
	TCK	Logic input	TAP clock
	TDO	Logic output	TAP data output

2.3 GPIO Multiplexing

Not all interfaces can be active at the same time. To provide flexibility, these shared interfaces are multiplexed with GPIOs.

Note: All 25 functional IOs come up as Function 0 at boot. JTAG is default enabled (as part of Function 0) instead of GPIO[23:19]. FW is responsible for enabling proper configuration later on.

Table 3. Multiplexed Functions

Function0	Function1	Function2
GPIO[3:0]	ADC/COMP[3:0]	SPI_M (4 IO)
GPIO[5:4]	ADC/COMP[5:4]	RTC_CLK_OUT, SYS_CLK_OUT
GPIO[7:6]	ADC/COMP[7:6]	I2C (2 IO)
GPIO[11:8]	ADC/COMP[11:8]	SPI_S (4 IO)
GPIO[15:12]	ADC/COMP[15:12]	UART_A (4 IO)
GPIO[18:16]	ADC/COMP[18:16]	SPI_M (3 IO)
JTAG (5 IO)	GPIO[23:19]	UART_B (4 IO), PWM0
GPIO[24]	-	PWM1

3 Ballout and Package Information

The Intel® Quark™ microcontroller D2000 comes in 6 mm x 6 mm Quad Flat No-Leads (QFN) Package.

3.1 SoC Attributes

- Package parameters: 6 mm X 6 mm (QFN)
- Pad Count: 40

All Units: mm

Tolerances if not specified:

- .X: ± 0.1
- .XX: ± 0.05
- Angles: ± 1.0 degrees

3.2 Package Diagrams

Figure 2. Package Diagram QFN 40 pin (0.5mm pitch)

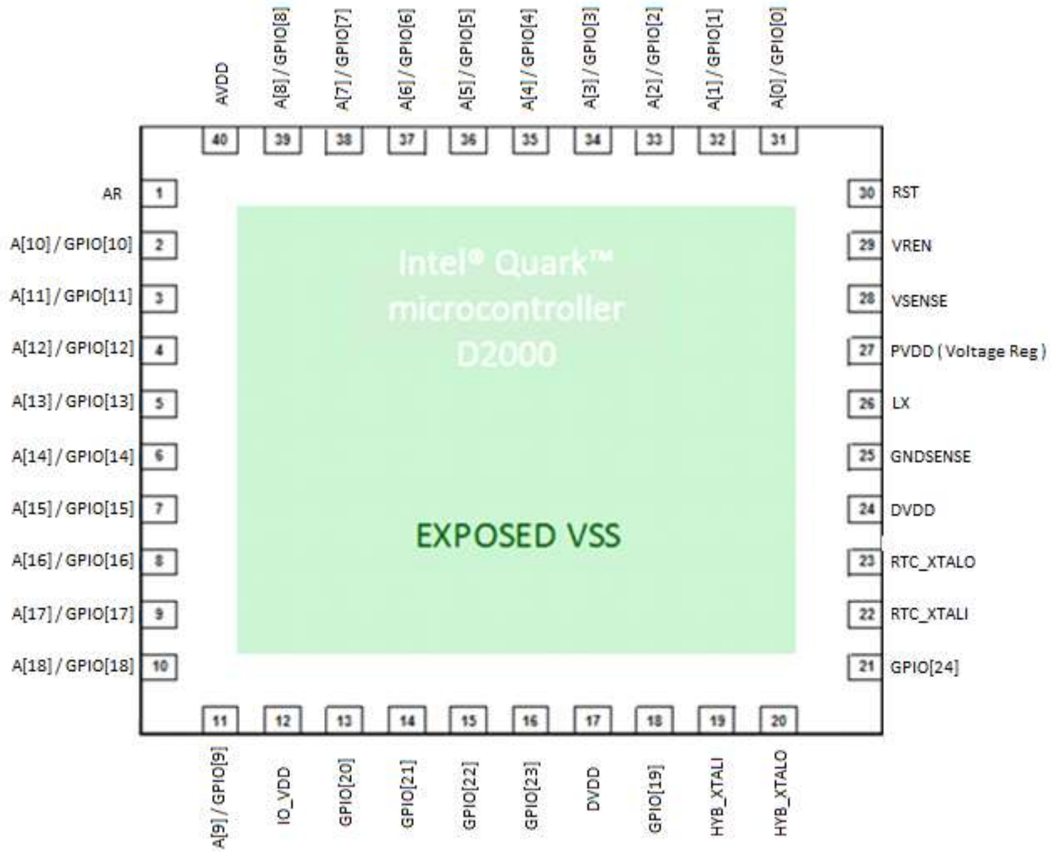
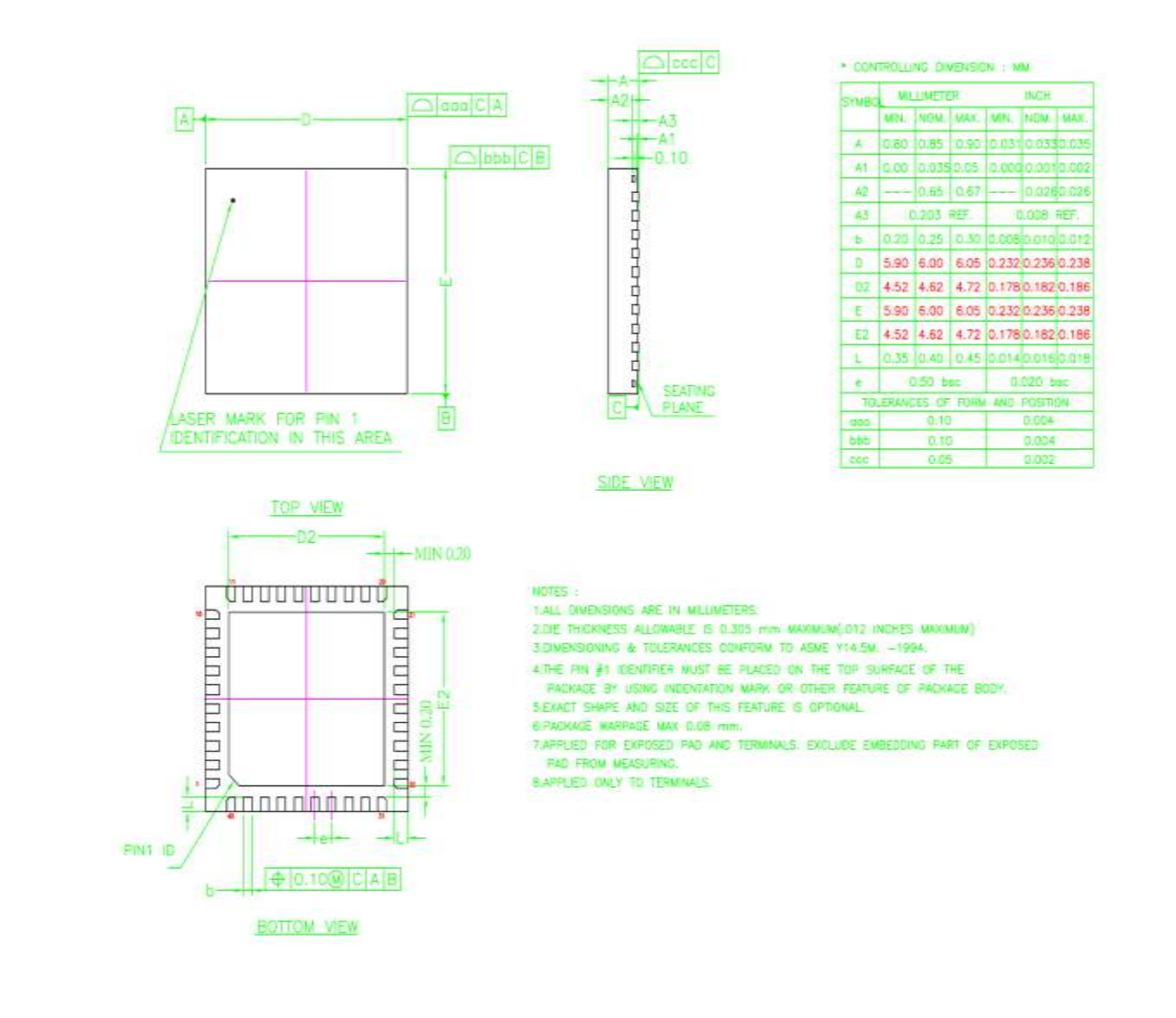


Figure 3. Mechanical Drawing of Package



* CONTROLLING DIMENSION : MM

SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.80	0.85	0.90	0.031	0.033	0.035
A1	0.00	0.035	0.05	0.000	0.001	0.002
A2	----	0.65	0.67	----	0.026	0.026
A3	0.203 REF.			0.008 REF.		
b	0.20	0.25	0.30	0.008	0.010	0.012
D	5.90	6.00	6.05	0.232	0.236	0.238
D2	4.52	4.62	4.72	0.178	0.182	0.186
E	5.90	6.00	6.05	0.232	0.236	0.238
E2	4.52	4.62	4.72	0.178	0.182	0.186
L	0.35	0.40	0.45	0.014	0.016	0.018
e	0.50 bsc			0.020 bsc		
TOLERANCES OF FORM AND POSITION						
aaa	0.10			0.004		
bbb	0.10			0.004		
ccc	0.05			0.002		

NOTES :

1.ALL DIMENSIONS ARE IN MILLIMETERS.

2.DIE THICKNESS ALLOWABLE IS 0.305 mm MAXIMUM(.012 INCHES MAXIMUM)

3.DIMENSIONING & TOLERANCES CONFORM TO ASME Y14.5M. -1994.

4.THE PIN #1 IDENTIFIER MUST BE PLACED ON THE TOP SURFACE OF THE PACKAGE BY USING INDENTATION MARK OR OTHER FEATURE OF PACKAGE BODY.

5.EXACT SHAPE AND SIZE OF THIS FEATURE IS OPTIONAL.

6.PACKAGE WARPAGE MAX 0.08 mm.

7.APPLIED FOR EXPOSED PAD AND TERMINALS. EXCLUDE EMBEDDING PART OF EXPOSED PAD FROM MEASURING.

8.APPLIED ONLY TO TERMINALS.

3.3 Pin Multiplexing

There are 15 dedicated pins + 1 QFN GND plane and 25 functional pins which can be configured as GPIO (GPIO[24:0]) or other functions (I^2C/UART/SPI/JTAG). There are two major IO modes: user mode and test mode. In user mode, each pin can be individually configured in one of the 4 user modes (FUNC 0/1/2/3). By default, after power-on-reset (RST_N) or cold reset, SoC comes up in user mode 0 function (FUNC0). SOC firmware/software is responsible for enabling the platform specific configuration by programming the respective IO pad control registers.

Out of 25 functional pins, 19 pins are double bonded to Analog input pads and digital pads while 6 pins are digital only pads. All analog pads (AI[18:0]) are specified with respect to AVDD and all digital pads are with respect to IOVDD. The analog inputs (AI) are connected to ADC or Comparators inside the SoC. AI[5:0] is connected to fast response/high performance comparator / fast channel of ADC; and AI[18:6] are connected to slow response/low power comparator / slow channel of ADC. Any wake capable analog inputs shall be connected only to any of AI[18:6] and not to AI[5:0].

Table 4. Pin Multiplexing

Pin Number	Pin/Ball Name	Type	Voltage	Function 0	Function 1	Function 2	Function 3
VSS	GND	PWR	0 V	GND	GND	GND	GND
27	PVDD	PWR	PVDD	PVDD	PVDD	PVDD	PVDD
40	AVDD	PWR	PVDD	AVDD	AVDD	AVDD	AVDD
12	IOVDD	PWR	PVDD	IOVDD	IOVDD	IOVDD	IOVDD
28	VSENSE	PWR	DVDD	VSENSE	VSENSE	VSENSE	VSENSE
25	GSENSE	PWR	0 V	GSENSE	GSENSE	GSENSE	GSENSE
26	LX	PWR	DVDD	LX	LX	LX	LX
29	VREN	PWR	PVDD	VREN	VREN	VREN	VREN
17	DVDD	PWR	DVDD	DVDD	DVDD	DVDD	DVDD
30	RST_N	RST	PVDD	RST_N	RST_N	RST_N	RST_N
22	RTC_XTALI	CLK	DVDD	RTC_XTALI	RTC_XTALI	RTC_XTALI	RTC_XTALI
23	RTC_XTALO	CLK	DVDD	RTC_XTALO	RTC_XTALO	RTC_XTALO	RTC_XTALO
19	HYB_XTALI	CLK	DVDD	HYB_XTALI	HYB_XTALI	HYB_XTALI	HYB_XTALI
20	HYB_XTALO	CLK	DVDD	HYB_XTALO	HYB_XTALO	HYB_XTALO	HYB_XTALO
1	AR	PWR	AVDD	AR	AR	AR	AR
31	F_0	GPIO	IOVDD/ AVDD	GPIO0	AI0	SPI_M_SS0	
32	F_1	GPIO	IOVDD/ AVDD	GPIO1	AI1	SPI_M_SS1	
33	F_2	GPIO	IOVDD/ AVDD	GPIO2	AI2	SPI_M_SS2	
34	F_3	GPIO	IOVDD/ AVDD	GPIO3	AI3	SPI_M_SS3	
35	F_4	GPIO	IOVDD/ AVDD	GPIO4	AI4	RTC_CLK_OUT	

Pin Number	Pin/Ball Name	Type	Voltage	Function 0	Function 1	Function 2	Function 3
36	F_5	GPIO	IOVDD/ AVDD	GPIO5	AI5	SYS_CLK_OUT	
37	F_6	GPIO	IOVDD/ AVDD	GPIO6	AI6	I2C_SCL	
38	F_7	GPIO	IOVDD/ AVDD	GPIO7	AI7	I2C_SDA	
39	F_8	GPIO	IOVDD/ AVDD	GPIO8	AI8	SPI_S_SCLK	
11	F_9	GPIO	IOVDD/ AVDD	GPIO9	AI9	SPI_S_SDIN	
2	F_10	GPIO	IOVDD/ AVDD	GPIO10	AI10	SPI_S_SDOUT	
3	F_11	GPIO	IOVDD/ AVDD	GPIO11	AI11	SPI_S_SCS	
4	F_12	GPIO	IOVDD/ AVDD	GPIO12	AI12	UART_A_TXD	
5	F_13	GPIO	IOVDD/ AVDD	GPIO13	AI13	UART_A_RXD	
6	F_14	GPIO	IOVDD/ AVDD	GPIO14	AI14	UART_A_RTS/UART_A_DE	
7	F_15	GPIO	IOVDD/ AVDD	GPIO15	AI15	UART_A_CTS/UART_A_RE	
8	F_16	GPIO	IOVDD/ AVDD	GPIO16	AI16	SPI_M_SCLK	
9	F_17	GPIO	IOVDD/ AVDD	GPIO17	AI17	SPI_M_TXD	
10	F_18	GPIO	IOVDD/ AVDD	GPIO18	AI18	SPI_M_RXD	-
18	F_19	GPIO	IOVDD	TDO	GPIO19	PWM0	-
13	F_20	GPIO	IOVDD	TRST_N	GPIO20	UART_B_TXD	-
14	F_21	GPIO	IOVDD	TCK	GPIO21	UART_B_RXD	-
15	F_22	GPIO	IOVDD	TMS	GPIO22	UART_B_RTS/UART_B_DE	-
16	F_23	GPIO	IOVDD	TDI	GPIO23	UART_B_CTS/UART_B_RE	-
21	F_24	GPIO	IOVDD	GPIO24	-	PWM1	-
24	DVDD_2	PWR	DVDD_2	DVDD_2	DVDD_2	DVDD_2	DVDD_2

All Analog IOs (AI[18:0]/ADC[18:0]) are with respect to AVDD rail.

All Digital IOs are 3.3V tolerant.

All Digital IO include configurable drive strength namely low-drive (12 mA) and high-drive (16 mA) modes. By default, all digital IOs come up in low-drive mode.

All Digital IO include a configurable pullup with pull-up disabled by default, except for F_20, F_22, F_23 (TRST_N, TMS, TDI)) where pull-up is enabled by default.

UART_A/B_CTS (input) or UART_A/B_RE (output) is available based on UART mode of operation (RS232 or RS485) configurable in respective UART controller. Similarly for UART_A/B_RTS (output) or UART_A/B_DE (output). Default is RS232. FW can choose to enable both input and output direction for CTS/RE pins and hardware will control the output enable of these pins based on RS232 or RS485 mode of operation if these pins are configured for UART function.

JTAG interface is default enabled (user mode 0) to assist in debug as well as embedded flash programming.

There is a F_25 pad which is not bonded to any pin. This pad (mapped as pin 25 for pinmux configuration in System Control Subsystem) shall be configured by FW in input disabled mode (Input Enable = 0; User mode 3 which makes Output Enable disabled).

DVDD_2 and DVDD are identical and are to be connected to the same source.

3.4 Alphabetical Ball Listing

Table 5 shows pins arranged in increasing order of pin number.

Table 5. Pins Listed in increasing Order of Pin Number

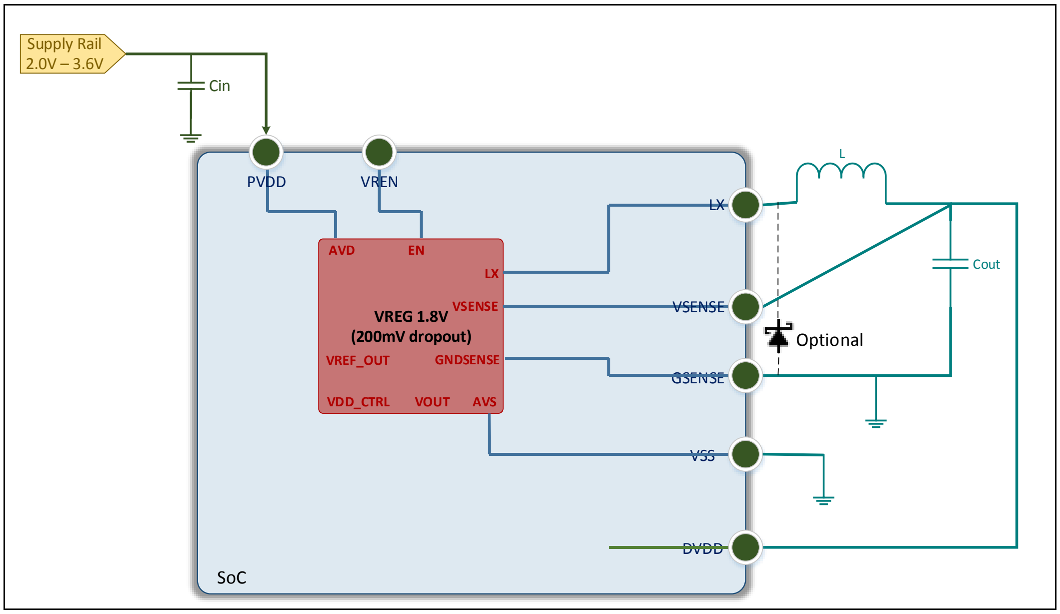
Pin Number	Pin/Ball Name	Type	Voltage	Function 0	Function 1	Function 2	Function 3
1	AR	PWR	AVDD	AR	AR	AR	AR
2	F_10	GPIO	IOVDD/AVDD	GPIO10	AI10	SPI_S_SDOUT	
3	F_11	GPIO	IOVDD/AVDD	GPIO11	AI11	SPI_S_SCS	
4	F_12	GPIO	IOVDD/AVDD	GPIO12	AI12	UART_A_TXD	
5	F_13	GPIO	IOVDD/AVDD	GPIO13	AI13	UART_A_RXD	
6	F_14	GPIO	IOVDD/AVDD	GPIO14	AI14	UART_A_RTS/UART_A_DE	
7	F_15	GPIO	IOVDD/AVDD	GPIO15	AI15	UART_A_CTS/UART_A_RE	
8	F_16	GPIO	IOVDD/AVDD	GPIO16	AI16	SPI_M_SCLK	
9	F_17	GPIO	IOVDD/AVDD	GPIO17	AI17	SPI_M_TXD	
10	F_18	GPIO	IOVDD/AVDD	GPIO18	AI18	SPI_M_RXD	-
11	F_9	GPIO	IOVDD/AVDD	GPIO9	AI9	SPI_S_SDIN	
12	IOVDD	PWR	PVDD	IOVDD	IOVDD	IOVDD	IOVDD
13	F_20	GPIO	IOVDD	TRST_N	GPIO20	UART_B_TXD	-
14	F_21	GPIO	IOVDD	TCK	GPIO21	UART_B_RXD	-
15	F_22	GPIO	IOVDD	TMS	GPIO22	UART_B_RTS/UART_B_DE	-
16	F_23	GPIO	IOVDD	TDI	GPIO23	UART_B_CTS/UART_B_RE	-
17	DVDD	PWR	DVDD	DVDD	DVDD	DVDD	DVDD
18	F_19	GPIO	IOVDD	TDO	GPIO19	PWM0	-
19	HYB_XTALI	CLK	DVDD	HYB_XTALI	HYB_XTALI	HYB_XTALI	HYB_XTALI
20	HYB_XTALO	CLK	DVDD	HYB_XTALO	HYB_XTALO	HYB_XTALO	HYB_XTALO
21	F_24	GPIO	IOVDD	GPIO24	-	PWM1	-
22	RTC_XTALI	CLK	DVDD	RTC_XTALI	RTC_XTALI	RTC_XTALI	RTC_XTALI

Pin Number	Pin/Ball Name	Type	Voltage	Function 0	Function 1	Function 2	Function 3
23	RTC_XTALO	CLK	DVDD	RTC_XTALO	RTC_XTALO	RTC_XTALO	RTC_XTALO
24	DVDD_2	PWR	DVDD_2	DVDD_2	DVDD_2	DVDD_2	DVDD_2
25	GSENSE	PWR	0 V	GSENSE	GSENSE	GSENSE	GSENSE
26	LX	PWR	DVDD	LX	LX	LX	LX
27	PVDD	PWR	PVDD	PVDD	PVDD	PVDD	PVDD
28	VSENSE	PWR	DVDD	VSENSE	VSENSE	VSENSE	VSENSE
29	VREN	PWR	PVDD	VREN	VREN	VREN	VREN
30	RST_N	RST	PVDD	RST_N	RST_N	RST_N	RST_N
31	F_0	GPIO	IOVDD/AVDD	GPIO0	AI0	SPI_M_SS0	
32	F_1	GPIO	IOVDD/AVDD	GPIO1	AI1	SPI_M_SS1	
33	F_2	GPIO	IOVDD/AVDD	GPIO2	AI2	SPI_M_SS2	
34	F_3	GPIO	IOVDD/AVDD	GPIO3	AI3	SPI_M_SS3	
35	F_4	GPIO	IOVDD/AVDD	GPIO4	AI4	RTC_CLK_OUT	
36	F_5	GPIO	IOVDD/AVDD	GPIO5	AI5	SYS_CLK_OUT	
37	F_6	GPIO	IOVDD/AVDD	GPIO6	AI6	I2C_SCL	
38	F_7	GPIO	IOVDD/AVDD	GPIO7	AI7	I2C_SDA	
39	F_8	GPIO	IOVDD/AVDD	GPIO8	AI8	SPI_S_SCLK	
40	AVDD	PWR	PVDD	AVDD	AVDD	AVDD	AVDD
VSS	GND	PWR	0 V	GND	GND	GND	GND

3.5 Platform Requirements

3.5.1 3.5.1 Internal Voltage Regulator

Figure 4. Internal Voltage Regulator



The requirement for external component at PCB level is as shown in the following table:

Component Name	Description		Characteristic	Value	Accuracy	Unit
Cin	Input Capacitor		Ceramic	470	+/- 20%	nF
Cout	Output Tank Capacitor		Ceramic	4.7	+/- 20%	µF
L	Switching Inductor			47	+/- 30%	µH

ESR rating for Cin and Cout:

For Cout: Resr(TYP) = 100mOhm / Resr(MAX) = 500mOhm

For Cin: Resr(TYP) = 25mOhm / Resr(MAX) = 100mOhm

Please note for Cout, the higher ESR is, the higher the ripple is.

DCR rating for on-board inductor

L: DCR(TYP) = 100mOhm, DCR(MAX) = 500mOhm

The DCR impacts the efficiency of the regulator. The higher the DCR is, the lower the efficiency is.

Also, the voltage drop across the DCR increases the min dropout the regulator can support. For example, the minimum dropout specified in the spec is 200mV by considering a 0.1 Ohm DCR. If the DCR is higher, the dropout will also be higher.

Table 6. Parasitic Requirement for Voltage Regulator Pins

Item	Description
PVDD	Maximum total resistance (including wirebond, package pin and board routing) between PVDD PIN and Input Supply must be less than 400mΩ/nb PVDD pads.
VSENSE	Has to be star-connected: no current should flow through this path between VSENSE pin and the regulated voltage node. Access resistance to this port (including wirebond, package pin and board routing) must be lower than 100Ω.
GNDSENSE	Maximum total resistance (including wirebond, package pin and board routing) between GNDSENSE and Ground plane must be less than 400mΩ. This pin should be star connected to the reference (GND) of the load circuit. No load current should flow through this connection.
AVS	Maximum total resistance (including wirebond, package pin and board routing) between AVS and Ground plane must be less than 400mΩ/nb AVS pads. No other connections allowed on this PAD.
VDD_CTRL	Maximum total resistance (including wirebond, package pin and board routing) between VDD_CTRL pin and Input Supply must be less than 20Ω. May be tied to VREG external node, VOUT output or any other supply
DVDD	As small as possible
LX	Maximum total resistance (including wirebond, package pin and board routing) between LX PIN and package Input Pin must be less than 400mΩ/nb LX pins.
VOUT	Maximum total resistance (including internal wiring parasitic, wirebond, package pin and board routing) between VREG pin and the regulation point (VSENSE pin connection) must be less than 20Ω.
C_{IN}	Must be placed within 0.2-0.4 inch of IC
C_{out}	Must be placed within 0.2-0.4 inch of IC
Package-Board Gnd	Inductance between package gnd plane and board gnd plane < 1nH

When internal voltage regulator is disabled (VR_EN = 0), PVDD must be powered up, GNDSENSE and AVS must be grounded. LX is HIZ, VSENSE to be grounded.

3.5.2 RTC Oscillator

For 32.768 kHz RTC Oscillator, refer to http://www.murata.com/products/timingdevice/crystalu/technical/notice for PCB guidelines.

Load capacitor on XTAL pins are integrated inside the chip but they are not adjustable. A nominal value of 11 pF is on each pin (range: 6 pF minimum and 17 pF maximum).

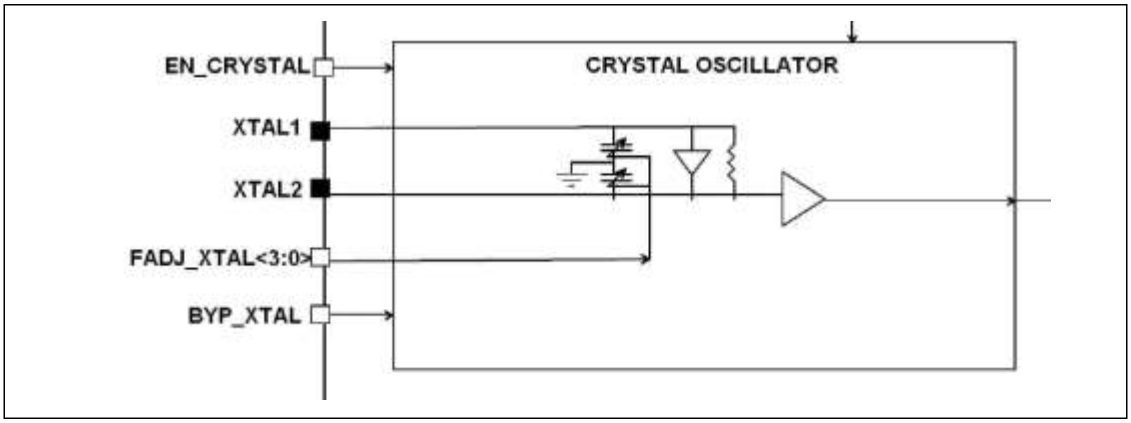
If there is no RTC Oscillator need, then a platform need not mount RTC XTAL on board and keep RTC_XTALI, RTC_XTALO pins grounded.

3.5.3 Hybrid Oscillator

The hybrid oscillator can work in internal Silicon RC oscillator mode with +/- 2% accuracy (after trimming). If a system does not require a high accuracy system clock, then HYBOSC 32MHz XTAL need not be mounted on board to save cost and in such case, keep HYB_XTALI, HYB_XTALO pins as no-connect (floating).

A programmable capacitive load can be added on the crystal terminals internal to the chip to fine tune the crystal frequency. The range and step size will vary depending on external load, process corners and crystal parameters. Load capacitance is adjustable through SCSS register in SoC (OSC0_CFG1.OSC0_FADJ_XTAL[3:0] register – range supported is 5.55 pF to 15.03 pF with default value of 10pF).

Figure 5. Typical Configuration of Hybrid Oscillator



3.5.4 ADC

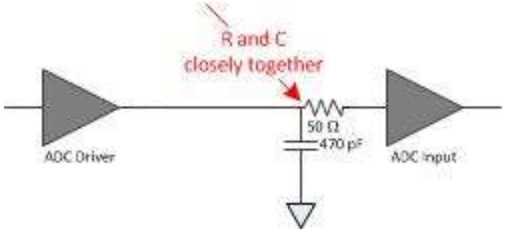
A simple filter is recommended to be put on board on Analog Inputs [19:0] connected to ADC, to reduce external noise. It should be understood that any additional frequency signals within the band of interest will be present in the output spectrum contributing for a performance impact. Additionally the input signal is disturbed by the fast transients due to the fast charging of the ADC input capacitor during the start of the sampling period. These transients should vanish quite rapidly and the voltage across the ADC input capacitance should reach a steady state very fast. Nevertheless in case the ADC input network is not correctly terminated these transients can travel back and forth through the input line between the ADC driver and ADC input generating a ringing. This will lead to an incorrect sampling of input signal.

A recommended input scheme to prevent this possible consequence can be the following:

- 50 ohm resistor (in ADC driver side). The resistor level should match the characteristic impedance of PCB trace
- 470pF capacitor to ground (in ADC driver side)

In order that they are effective, resistor and capacitor must be connected very close to each other and placed towards the ADC Analog input (SoC side), as shown in Figure 6.

Figure 6. Effective Placement of Resistor and Capacitor



Power supply decoupling on AVDD rail input shall be done with 1uF || 10 nF decoupling capacitors. The 10 nF capacitor shall be ceramic (good quality) and must be placed as close as possible to the SoC.

4 Electrical Characteristics

4.1 Thermal Specifications

Ambient temperature = -40°c to +85°c.

4.2 Voltage and Current Specifications

4.2.1 Absolute Maximum Ratings

Table 7. Absolute Maximum Voltage Ratings

Symbol	Ratings	min	Max	Unit	Notes
PVDD	Battery Supply	2.0	3.63	V	• Used by internal voltage regulator. • If internal VR is enabled, current draw is 55 mA max (specification limit). Actual current draw would be less based on usecase scenario. • If external VR feeds DVDD, then current draw on PVDD is very low (< 5 µA).
AVDD	Analog Supply	2.0	3.63	V	• Can be same source as PVDD but noise isolated. • Used by ADC and Comparators. • Current draw is 2 mA max.
IOVDD	Digital IO Supply	1.62	3.63	V	• Can be same source as PVDD but optionally noise isolated. • Used by 26 IO digital pads. • Current draw depends on platform components and how many digital IO pads are used. • Each IO pad can source 14mA/18mA max. So max current draw by all IO pads = 500 mA. But actual consumption would be much less.

Symbol	Ratings	min	Max	Unit	Notes
DVDD/DVDD_2	Regulated Core Voltage	1.62	1.98	V	• Core voltage domain. Only 1 Always-on rail for entire SoC core. This is regulated voltage. • Connected to external VR (if VREN=0); or to Internal VR output (if VREN=1)(post LC circuit on LX output pin). If external voltage regulator, current draw is 50 mA max.
DVDD/DVDD_2	Retention mode 1.35V during Deep Sleep Power State	1.2	1.43	V	• In Deep Sleep Power State, there is an option to lower core voltage to 1.35V (nominal) +/- 10% to further reduce sleep/leakage power, with implication of increased power state transition entry/exit latency.

Refer to Chapter 8, “Power Management” for information about voltage rail sequencing in internal voltage regulator mode and external voltage regulator mode.

4.3 Crystal Specifications

Figure 7. Crystal Specifications for Hybrid Oscillator

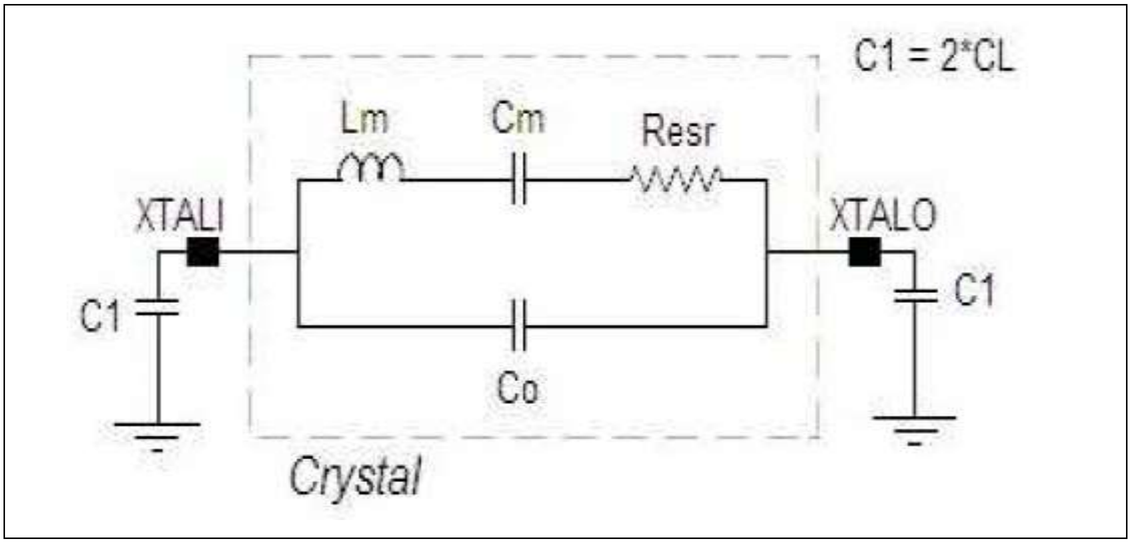


Table 8. 32MHz Crystal Oscillator specification

Symbol	Parameter	Min	Typ	Max	Unit
Fo	Crystal frequency	32	32	32	MHz
Cesr	Crystal ESR	12.68	14.41	50	Ω

Symbol	Parameter	Min	Typ	Max	Unit
Cm	Crystal Motional Cap	3.34		3.54	pF
Co	Crystal Shunt Cap	0.84		1.5	pF
CL	Crystal Load Cap		10		pF
Ftol	Frequency Tolerance	-30		30	ppm
Dlev	Drive Level (25Ω)			10	µW
Output clock frequency accuracy over PVT	Without crystal (si osc mode) after trim done at typical temperature	-2		+2	%
	With crystal before trim (excluding crystal frequency tolerance)	-100		100	ppm
	With crystal after trim	-50		80	ppm
Startup time	Si osc mode – frequency setting within 2% accuracy		2		µs
	Crystal mode – frequency settling within 100ppm accuracy		2000		µS

Table 9. 32kHz Crystal Oscillator specification

Symbol	Parameter	Min	Typ	Max	Unit
Fo	Crystal frequency		32,768		Hz
Cesr	Crystal ESR		50	80	KΩ
Cm	Crystal Motional Cap		3.7		pF
Co	Crystal Shunt Cap		1.2		pF
CL	Crystal Load Cap		7		pF
Ftol	Frequency Tolerance	-20		20	ppm
Dlev	Drive Level			1	µW

4.4 DC Specifications

4.4.1 IO DC specifications

Table 10. For IOVDD=3.3V

Symbol	Parameter	Min	Typ	Max	Unit
VIL	Input Low Voltage	-0.3		0.8	V
VIH	Input High Voltage	2		3.6	V
VOL	Output Low Voltage			0.4	V
VOH	Output High Voltage	2.4			V
IOL	12mA @ VOL	14.1	22.9	31.8	mA
	16mA @ VOL	18.8	30.6	42.4	mA
IOH	12mA @ VOH	20.7	41.9	69.6	mA
	16mA @ VOH	28.7	58.2	96.7	mA
RPU	Pull-up Resistor	34K	49K	74K	Ohm
VT	Threshold Point	1.33	1.4	1.47	V
VT+	Schmitt Trigger L-> H Threshold Point	1.53	1.6	1.66	V
VT-	Schmitt Trigger H-> L Threshold Point	1.13	1.2	1.27	V

Table 11. For IOVDD=1.8V

Symbol	Parameter	Min	Typ	Max	Unit
VIL	Input Low Voltage	-0.3		0.63	V
VIH	Input High Voltage	1.17		3.6	V
VOL	Output Low Voltage			0.45	V
VOH	Output High Voltage	1.35			V
IOL	12mA @ VOL	5.8	12.1	21.6	mA
	16mA @ VOL	7.7	16.2	28.7	mA
IOH	12mA @ VOH	4.7	12.0	24.3	mA
	16mA @ VOH	6.6	16.7	33.8	mA
RPU	Pull-up Resistor	69K	115K	201K	Ohm
VT	Threshold Point	0.82	0.89	0.93	V

Symbol	Parameter	Min	Typ	Max	Unit
VT+	L-> H Threshold Point	0.99	1.07	1.12	V
VT-	H-> L Threshold Point	0.62	0.69	0.77	V

4.4.2 Undershoot Voltage Support

SoC supports a undershoot voltage of 300 mV (VIL min of -0.3V) on its input pins.

4.4.3 ADC IO DC characteristics

Table 12. ADC IO DC Characteristics

Symbol	Parameter	Min	Typ	Max	Unit
Full-scale input range		0		3.63	V
VREFP	Positive reference voltage	2	AVDD	AVDD	V
AGNDREF	Negative reference voltage	0	0	0.1	V
ADC_cap	Input sampling capacitance		5		pF

4.5 System Power Consumption

Scenario	Condition	Min	Typ	Max	Unit
Total Active Power (Pavdd + Ppvdd + Piovdd) with Internal VR enabled. - CPU running Coremark benchmark workload - All peripherals clock gated - Hybrid Oscillator and RTC Oscillator running - ADC and Comparators powered down	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 32 MHz, $-40 \le T \le 85^{0}$c	-	28.35	28.88	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 16 MHz, $-40 \le T \le 85^{0}$c	-	16.79	17.70	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 8 MHz, $-40 \le T \le 85^{0}$c	-	8.18	9.17	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 4 MHz, $-40 \le T \le 85^{0}$c	-	4.60	4.98	mW

Scenario	Condition	Min	Typ	Max	Unit
Total Standby/Halt Power (Ppvdd + Pavdd + Piovdd) with Internal VR enabled. - CPU in C2 power state (executed HALT instruction) - All peripherals clock gated - Hybrid Oscillator and RTC Oscillator running - ADC and Comparators powered down	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 32 MHz, -40≤T≤85^{0}c	-	3.24	3.93	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 16 MHz, -40≤T≤85^{0}c	-	1.99	2.04	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 8 MHz, -40≤T≤85^{0}c	-	1.19	1.21	mW
	Vpvdd=Vavdd=Viovdd=3.3V fCPU = 4 MHz, -40≤T≤85^{0}c	-	0.90	0.93	mW
Total Deep Sleep RTC Current (Ipvdd + Iavdd + Iiovdd) with AON Periodic Timer Wake. - CPU in C2 power state (executed HALT instruction) - All peripherals clock gated - Hybrid Oscillator powered down and RTC Oscillator running - ADC and Comparators powered down - Internal Voltage Regulator enabled in Linear Regulator mode (1.8V or 1.35V voltage output)	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=25*c	-	3.37	3.90	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=-40*c	-	2.73	3.28	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=85*c	-	6.63	10.38	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=25^{0}c	-	2.47	3.23	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=-40^{0}c	-	1.89	2.14	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=85^{0}c	-	5.10	8.93	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=25*c	-	2.36	2.97	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=-40*c	-	1.49	2.15	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=85*c	-	5.20	9.90	uA

Scenario	Condition	Min	Typ	Max	Unit
Total Deep Sleep NoRTC Current (Ipvdd + Iavdd + Iiovdd) with Low Power Comparator Wake. - CPU in C2 power state (executed HALT instruction) - All peripherals clock gated - Hybrid Oscillator and RTC Oscillator powered down - ADC and 18 Comparators powered down. 1 Low Power Comparator enabled for wake. - Internal Voltage Regulator enabled in Linear Regulator mode (1.8V or 1.35V voltage output)	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=25*c	-	1.71	2.36	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=-40*c	-	1.27	1.48	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=85*c	-	4.54	8.20	uA
Total Deep Sleep NoRTC Current (Ipvdd + Iavdd + Iiovdd) with GPIO Wake. - CPU in C2 power state (executed HALT instruction) - All peripherals clock gated - Hybrid Oscillator and RTC Oscillator powered down - ADC and Comparators powered down. - 1 GPIO input enabled for level sensitive interrupt wake - Internal Voltage Regulator enabled in Linear Regulator mode (1.8V or 1.35V voltage output)	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=25*c	-	1.82	2.96	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=-40*c	-	1.30	1.65	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.8V, T=85*c	-	4.93	9.16	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=25*c	-	1.35	1.71	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=-40*c	-	0.94	1.26	uA
	Vpvdd=Vavdd=Viovdd=3.3V Vdvdd = 1.35V, T=85*c	-	3.95	7.82	uA

4.6 AC Specifications

4.6.1 SPI Master IO AC characteristics

SPI Master interface consists of:

- Outputs SPI_M_SCLK, SPI_M_TXD, SPI_M_SS[3:0]
- Inputs SPI_M_RXD

The interface is timed with respect to SPI_M_SCLK which is output from SoC. A given signal is launched with respect of a configured edge and sampled with respect to the opposite edge. Thus it is a half-cycle setup/hold path.

Parameter	Min	Max
SPI Master Clock SPI_M_SCLK frequency	-	16 MHz (= sysclk / 2).
Output delay for SPI_M_SS[3:0]/SPI_M_TXD with respect to SPI_M_SCLK launch clock edge	0 ns	14 ns
Setup time of SPI_M_RXD with respect to SPI_M_SCLK sampling edge	15 ns	-
Hold time of SPI_M_RXD with respect to SPI_M_SCLK sampling edge	0 ns	-

Output load supported for SPI_M_TXD, SPI_M_SS[3:0], SPI_M_SCLK outputs is 25 pF max to support max rate of 16 Mbps.

4.6.2 SPI Slave IO AC characteristics

SPI Slave interface consists of:

- Outputs SPI_S_SDOUT
- Inputs SPI_S_SCLK, SPI_S_SDIN, SPI_S_SCS

As per SPI protocol, the interface is timed with respect to SPI_S_SCLK which is input to SoC. A given signal is launched with respect of a configured edge and sampled with respect to the opposite edge. Thus it is a half-cycle setup/hold path.

Output load supported for SPI_S_SDOUT output is 50 pF max and 5 pF min.

Parameter	Min	Max
SPI Slave Clock SPI_S_CLK Frequency	-	3.2 MHz (= 312.5 ns)
Setup time of SPI_S_SDIN/SPI_S_SCS with respect to sampling edge of SPI_S_SCLK input	-	30 ns
Hold time of SPI_S_SDIN/SPI_S_SCS with respect to sampling edge of SPI_S_SCLK input	-	120 ns

Parameter	Min	Max
Output delay of SPI_S_SDOUT with respect to launching edge of SPI_S_SCLK input	0 ns	120 ns

4.6.3 I^{2}C Master/Slave IO AC characteristics

I^{2}C interface consists of I2C_SCL and I2C_SDA bidirectional IOs and can operate in either master or slave mode. It can operate in standard mode (with data rates 0 to 100 Kbps), fast mode (with data rates less than or equal to 400 Kbps) or fast mode plus (with data rates less than or equal to 1 Mbps). To support fast mode plus, i2c_m0_clk (= system clock) shall be greater than or equal to 32 MHz.

I^{2}C specification dictates timing relationship between I2C_SCL and I2C_SDA to be met, which will be broadly taken care by the I^{2}C controller using configuration registers. SoC complies with the timing and load capacitance given in the I^{2}C specification.

4.6.4 General IO AC characteristics

Output load supported for all other IO outputs such as GPIO outputs, PWM, UART is 50 pF max and 5 pF min.

UART interface supports maximum baud rate of 2 Mbaud when system clock frequency is 32 MHz.

All GPIO and PWM outputs will get reflected within 1 system clock period of 30ns on the output pins.

4.6.5 JTAG Interface AC characteristics

The JTAG interface is a 5-pin interface timed with respect to TCK input clock.

- TMS, TDI are inputs which are sampled by SoC on rising edge of TCK and is expected to be driven by JTAG host on falling edge of TCK.
- TDO is output from SoC driven on falling edge of TCK and expected to be sampled by JTAG host on rising edge of TCK.
- TRSTB is asynchronous signal.

Delays shown in the following table are with respect to SoC.

Table 13. JTAG Interface AC characteristics

Parameter	Min	Max
TCK clock frequency	-	8 MHz (125 ns)
Setup time of TMS, TDI with respect to rising edge of TCK	25 ns	-
Hold time of TMS, TDI with respect to rising edge of TCK	0 ns	-
Output data valid delay of TDO with respect to falling edge of TCK	1 ns	40 ns

Parameter	Min	Max
Output tristate delay of TDO with respect to falling edge of TCK	-	40 ns

Output load supported for JTAG TDO output is 40 pF max and 5 pF min.

§

5 Register Access Methods

All SoC registers are accessed as Fixed Memory Mapped Registers.

The SoC does not contain any of these traditional x86 memory register types: Fixed IO, IO Referenced, Memory Referenced, PCI Configuration or Message Bus Registers.

5.1 Fixed Memory Mapped Register Access

Fixed Memory Mapped IO (MMIO) registers are accessed by specifying their 32-bit address in a memory transaction from the CPU core. This allows direct manipulation of the registers. Fixed MMIO registers are unmovable registers in memory space.

5.2 Register Field Access Types

Table 14. Register Access Types and Definitions

Access Type	Meaning	Description
RO	Read Only	In some cases, if a register is read only, writes to this register location have no effect. However, in other cases, two separate registers are located at the same location where a read accesses one of the registers and a write accesses the other register. See the I/O and memory map tables for details.
WO	Write Only	In some cases, if a register is write only, reads to this register location have no effect. However, in other cases, two separate registers are located at the same location where a read accesses one of the registers and a write accesses the other register. See the I/O and memory map tables for details.
R/W	Read/Write	A register with this attribute can be read and written.
R/WC	Read/Write Clear	A register bit with this attribute can be read and written. However, a write of 1 clears (sets to 0) the corresponding bit and a write of 0 has no effect.
R/WO	Read/Write-Once	A register bit with this attribute can be written only once after power up. After the first write, the bit becomes read only.
R/WLO	Read/Write, Lock-Once	A register bit with this attribute can be written to the non-locked value multiple times, but to the locked value only once. After the locked value has been written, the bit becomes read only.
/P	Sticky	A register bit with this attribute is sticky that is it will retain its state across warm-reset.
Reserved	Reserved	The value of reserved bits must never be changed.

Access Type	Meaning	Description
Default	Default	When the processor is reset, it sets its registers to predetermined default states. The default state represents the minimum functionality feature set required to successfully bring up the system. Hence, it does not represent the optimal system configuration. It is the responsibility of the system initialization software to determine configuration, operating parameters, and optional system features that are applicable, and to program the processor registers accordingly.

6 Mapping Address Spaces

The SoC supports a single flat Memory address space. The SoC does not support IO Address Space, PCI Configuration Space or Message Bus Space.

The D2000 processor core can directly access memory space either through code fetches over ITCM or data fetches over DTCM or through memory reads and writes over AHB fabric.

This chapter describes how memory space is mapped to interfaces and peripherals in the SoC.

6.1 Physical Address Space Mappings

Processor supports 32b addressing. There are 4 GB (32-bits) of physical address space that can be used as:

- Memory Mapped I/O (MMIO – I/O fabric)
- Physical Memory (System Flash/System SRAM)

The D2000 can access the full physical address space. DMA Controller, the only other master agent on AHB fabric, can only access regions of physical address space allowed via the multi-layer SoC fabric – see more details under SoC Fabric section.

All SoC peripherals map their registers and memory to physical address space. This chapter summarizes the possible mappings.

6.1.1 SoC Memory Map

The SoC Memory Map is divided up as follows:

- Processor Local APIC (LAPIC)
- I/O APIC
- SoC Configuration registers
- SoC Peripherals
- System Flash (Flash and ROM – implemented as 2 distinct regions of OTP)
- System SRAM (Internal)

The D2000 reset vector at 150h is located in OTP Code region of Intel® Quark™ microcontroller D2000. System addresses 0xFFFF_FFF0 and 0xFFFF_FFF8 are a special case by D2000 and get aliased to reset vector.

Table 15. SoC Memory Map

Function	Start Address	End Address	Size
LAPIC	0xFEE0_0000	0xFEE0_0FFF	4KB
IOAPIC	0xFEC0_0000	0xFECF_FFFF	1MB
SCSS (System Control Subsystem)	0xB080_0000	0xB080_3FFF	16KB
DMA	0xB070_0000	0xB070_0FFF	4KB
Internal SRAM Configuration	0xB040_0000	0xB040_03FF	1KB
Flash Configuration	0xB010_0000	0xB010_03FF	1KB
ADC	0xB000_4000	0xB000_43FF	1KB
I2C_0	0xB000_2800	0xB000_2BFF	1KB
UART_B	0xB000_2400	0xB000_27FF	1KB
UART_A	0xB000_2000	0xB000_23FF	1KB
SPI_S	0xB000_1800	0xB000_1BFF	1KB
SPI_M0	0xB000_1000	0xB000_13FF	1KB
GPIO	0xB000_0C00	0xB000_0FFF	1KB
APB Timer	0xB000_0800	0xB000_0BFF	1KB
RTC	0xB000_0400	0xB000_07FF	1KB
Watchdog Timer	0xB000_0000	0xB000_03FF	1KB
SRAM (DTCM & AHB)	0x0028_0000	0x0028_1FFF	8KB
OTP Data (AHB)	0x0020_0000	0x0020_0FFF	4KB
Flash Code (ITCM & AHB)	0x0018_0000	0x0018_7FFF	32KB
OTP Code (ITCM & AHB)	0x0000_0000	0x0000_1FFF	8KB

Notes:

- D2000 Reset Vector is located in OTP Code region at address 0x0000_0150
- All memory regions not covered in the SoC Memory Map are reserved. Reserved regions are unused. Reserved sections have been inserted to allow space for memory/peripheral address space to increase in derivative SoCs without re-arranging the address map. Access to Reserved regions trigger an interrupt to support debug of out of bound accesses. Writes to such regions have no effect while reads return a definite value depending on the interface.

AHB Fabric provides 4 HSELs to Flash Memory subsystem: Flash CREGs for configuration, Instruction Flash, OTP Code and OTP Flash Data.

Memory accesses are routed by the IO fabric based on fixed memory ranges that map to the SoC peripherals. These peripherals are: ADC, I^{2}C, UART*, SPI*, GPIO, APB Timer, RTC and Watchdog Timer. The fixed regions assigned to each peripheral are listed in the table above. See the register maps of all peripheral devices for details.

D2000 clock is gated by HW autonomously based on HALT detection. It is ungated by HW upon Interrupt assertion. There is an option to gate/ungate clock to Memory subsystem too with D2000 clock.

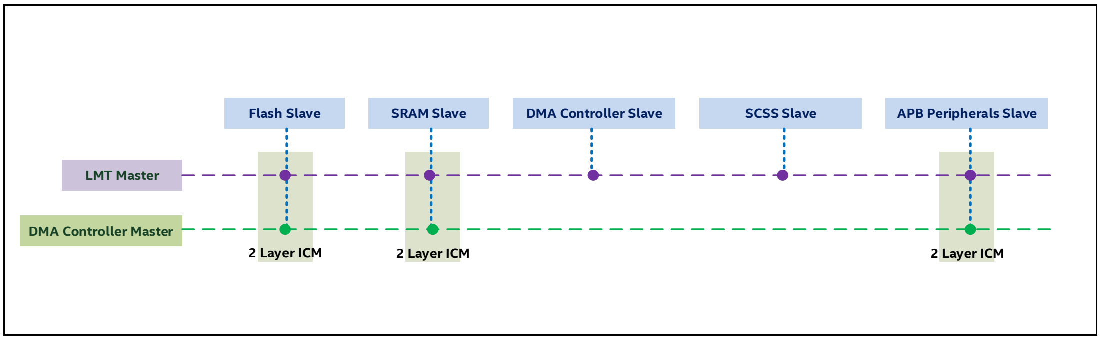
Other than D2000 and Memory subsystem clocks, for all other functions, SW controls gate/ungate of their respective clocks.

6.2 SoC Fabric

The SoC Fabric is a multi-layer AHB fabric that provides interconnect between 2 Masters and 5 Slaves.

The two masters are D2000 and DMA Controller – one on each AHB layer. The 5 slaves are: DMA Controller, Flash, SRAM, SCSS and APB peripherals. The multi-layer fabric allows multiple masters to access different slaves in parallel. When two or more masters try to access the same slave simultaneously, the slave arbitrates between the masters. With this topology, it is not possible for each master to access every slave connected to the SoC fabric. The D2000 master can access all the 5 slaves. DMA Controller master can only access Flash Slave, SRAM Slave and APB Peripherals Slave.

Figure 8. Multi-Layer AHB fabric with ICM



From an address decode perspective, DMA controller can target all destinations inside Flash subsystem (Flash Configuration registers, Instruction Flash, 8KB OTP, 4KB Data Flash) and all destinations inside SRAM subsystem (SRAM and SRAM Configuration registers).

The Multi-layer AHB fabric also makes use of the HMASTER port on two ICM components ICM_SRAM and ICM_FLASH. The HMASTER port on these two ICMs is used by the slave for memory protection of the SRAM and Flash slaves. Each HMASTER port for each layer on the ICM has a different ID code, this allows the slave to identify which master is currently trying to access memory.

Table 16. Multi-Layer AHB Fabric Master ID List

AHB Master ID Codes	
AHB Master	**ID Code**
D2000	0x1
DMA	0x4

APB Fabric has 10 slaves – GPIO, I^{2}C Master/Slave, SPI Master, SPI Slave, 2 UARTs, ADC, RTC, WDT and Timers block. Even though DMA engine can perform concurrent transfers, single outstanding transaction limitation of AHB prevents simultaneous transfers to/from multiple peripherals.

7 Clocking

The Intel® Quark™ microcontroller D2000 clocking is controlled by the Clock Control Unit (CCU). There are two primary clocks in Intel® Quark™ microcontroller D2000, a System clock and an RTC clock. The sources and frequencies of the primary clocks are described in subsequent sections. The CCU uses the primary clocks to generate secondary clocks to sub modules in the SoC. The secondary clocks are gated and scaled versions of the primary clocks.

7.1 Signal Descriptions

Table 17 provides the dependency on external signals/pins on clocking.

Table 17. External Signals

Signal Name	Direction/ Type	Description
External Crystal Interface		
HYB_XTALI	I/O	XTAL input or External system clock
HYB_XTALO	I/O	XTAL input
RTC_XTALI	I/O	RTC XTAL input
RTC_XTALO	I/O	RTC XTAL input
SYS_CLK_OUT	O	Divided 32 MHz System Clock output (pin multiplexed onto pin F_5 and not dedicated)
RTC_CLK_OUT	O	32 kHz RTC clock output (pin multiplexed onto pin F_4 and not dedicated)

7.2 Features

The CCU supports the following features:

- Supply of 32.768kHz RTC clock from internal crystal oscillator
- Supply of system clock from:
  - an internal hybrid oscillator which can work in either 4/8/16/32 MHz Silicon RC oscillator mode (+/- 2% clock accuracy) or 20-32 MHz Crystal oscillator mode (+/- 100 ppm clock accuracy; depending on crystal connected)
  - an external clock source (through HYB_XTALI input pin)
  - 32.768kHz RTC clock (internal muxing)
- Perform clock gating on all output clocks
- Perform clock scaling on all output clocks
- In “Low Power Compute” state of SoC, hybrid oscillator can be scaled down to 4 MHz with prescaler of 32 to get 125 kHz; or 32.768kHz (RTC clock wherein hybrid oscillator can be powered down).

In a minimalistic system configuration that does not require RTC clock or strict clock accuracy (< 100ppm) of system clock, such platforms can choose not to mount 32.768kHz XTAL and/or not to mount 32MHz XTAL and choose to work with 32MHz Silicon RC oscillator mode of hybrid oscillator. Intel® Quark™ microcontroller D2000 is designed to work with system clock only and does not mandate 32.768kHz RTC clock for power state transitions, if system/platform does not require RTC clock for its usecases.

7.2.1 System Clock - Hybrid Oscillator

When the system clock is sourced internally, a hybrid oscillator is used to generate the clock. The hybrid oscillator can be run in two modes, crystal or silicon mode depending on the frequency accuracy and current consumption requirements of the SoC. The hybrid oscillator contains the following features:

- Crystal mode
  - Generates 20-33 MHz clock
  - +/-100ppm (dependent on crystal frequency tolerance)
  - 2ms start-up time to reach +/-100ppm accuracy
- Silicon mode (default power up mode)
  - Generates 4/8/16/32 MHz clock
  - One time 10-bit factory trim
  - +/-20,000ppm (after process trim)
  - Temperature compensation block to limit frequency variation
  - 2us start-up time to reach +/-20,000ppm accuracy
- 4mA @ 32MHz in crystal mode [silicon characterization at ~650 uA @ 32 MHz]
- 450uA @ 32 MHz in silicon mode
- 300nA power down leakage current
- Operates with a supply range of 1.62-1.98V
- Outputs a rail-to-rail swing of 1.62-1.98V
- Power down mode to reduce power consumption within the SoC
- Glitch free mux for switching between hybrid and crystal oscillator clocks
- Bypass mode allows an external clock (fed through HYB_XTALI pin) to be provided through the hybrid oscillator. HYBOSC has to be configured in Crystal bypass mode (OSC0_EN_CRYSTAL=1, OSC0_EN_MODE_SEL=1, OSC0_BYP_XTAL=1)

7.2.2 RTC Oscillator

A Silicon Gate Oscillator is used to generate a 32.768kHz RTC clock. The RTC oscillator has the following features:

- Voltage operating range of 1.08-1.98V
- Accuracy of +/-20ppm
- Nominal current consumption of 150nA
- Power down mode consumption of 100nA

- Bypass mode allows an external clock (fed through RTC_XTALI pin) to be provided through the RTC oscillator. To achieve this, OSC1_BYP_XTAL_UP=1 and OSC1_PD=0)
- 600mS start-up time to reach +/-20ppm

The Intel® Quark™ microcontroller D2000 is designed to operate without RTC clock as well, if platform does not require RTC clock. RTC clock is needed for any of the following reasons:

- Periodic waking in low power sleep state.
- Real Time Clock
- GPIO based wake (input debouncing/filter)

If above reasons are not required in a platform, then RTC oscillator can be disabled without connecting RTC XTAL on board. In this mode, only comparator based wake can be enabled to exit low power state.

7.2.3 Root Clock Frequency Scaling

The Intel® Quark™ microcontroller D2000 supports a single root clock with multiple supported root clock frequencies

1. 32MHz high accuracy Crystal Oscillator – required for high accuracy applications.
2. 4/8/16/32MHz silicon Oscillator – A lower power operating mode used by applications that do not require a high frequency accuracy.
3. 32.768 kHz - entire SoC can operate out of 32.768 kHz clock as system clock (controlled by CCU_SYS_CLK_SEL register in SCSS) without enabling Hybrid Oscillator for such applications that require ultra low power without much compute performance.

7.2.4 Frequency Scaling

The Intel® Quark™ microcontroller D2000 supports a wide range of frequency scaling options to optimize power.

1. The root system clock frequency can be scaled to 4/8/16/32MHz
2. The Leaf peripheral clock can be independently scaled at /2 /4 /8 divisions

To apply a DFS setting the following procedure should be followed

1. Apply the clock divider value **CCU_XXX_CLK_DIV**
2. Apply the clock divider writing ‘0’ **CCU_XXX_CLK_DIV_EN**
3. Apply the clock divider writing ‘1’ **CCU_XXX_CLK_DIV_EN**

7.2.4.1 Peripheral DFS requirements

When using DFS it is the responsibility of firmware to adjust any settings in peripheral/timers to account for the frequency change. Example to achieve a UART baud rate of 115200 requires a different baud rate divider depending on the frequency.

7.2.4.2 Flash DFS requirements

When using DFS on the root fabric clock the flash wait states must be adjusted for both Flash instances. Refer to Memory Subsystem chapter for further details.

7.2.5 Dynamic Clock Gating

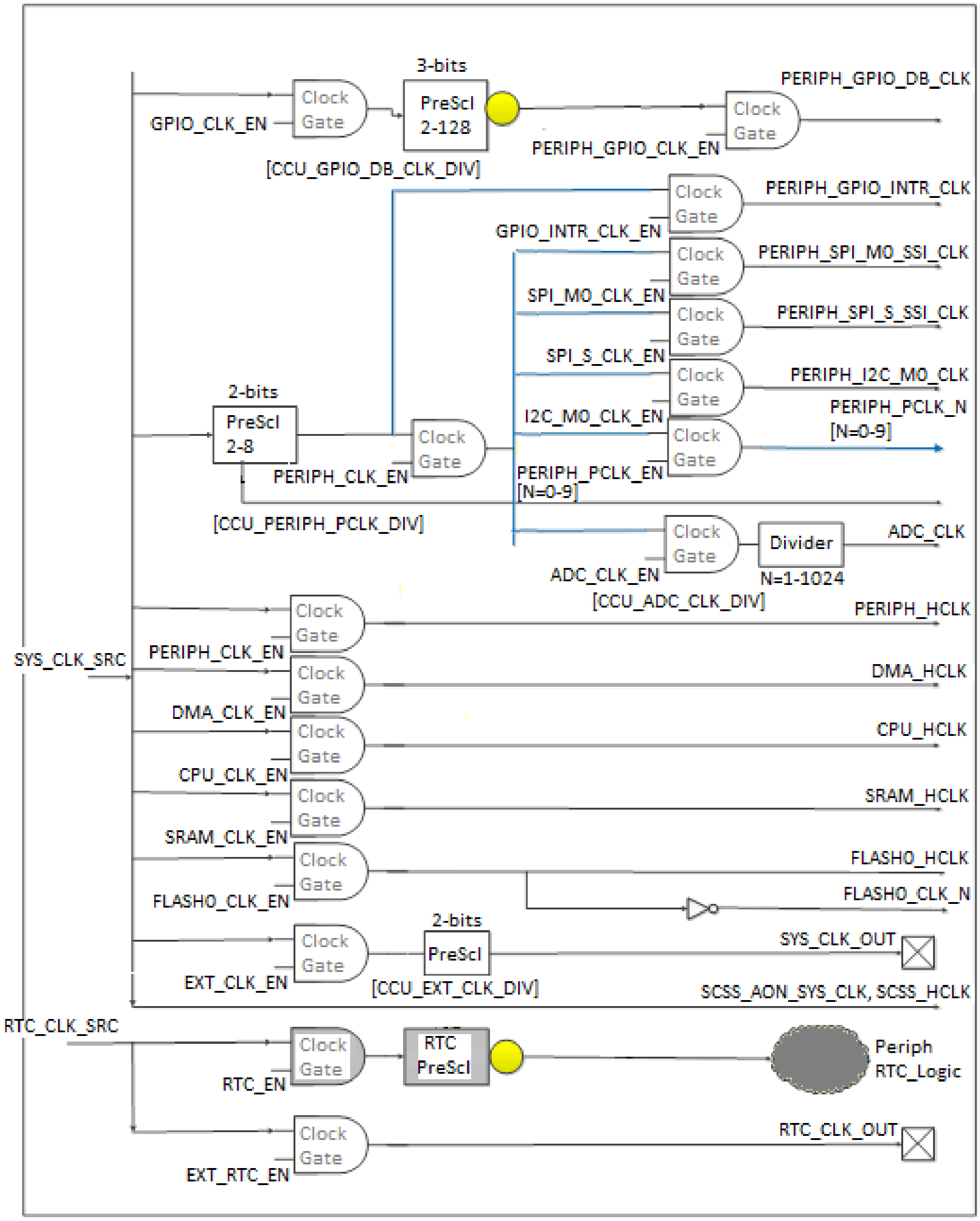
The Intel® Quark™ microcontroller D2000 supports a range of clock gating options

1. Each leaf clock can be dynamically gated by firmware

To apply a DCG the following procedure should be used.

1. Write ‘0’ to the clock gate register **CCU_XXX_PCLK_EN**
2. The following hardware clock gating options are supported
   – UART low power autonomous hardware clock gating
   – SPI low power autonomous hardware clock gating

Figure 9. CCU Clock Distribution



7.2.5.1 UART autonomous clock gating (ACG)

Both UART controllers support ACG mode (**CCU_UARTX_PCLK_EN_SW=0)**. ACG is asserted when the following occurs

1. Transmit and receive pipeline is clear (no data in the RBR/THR or TX/RX FIFO)
2. No activity has occurred on the SIN/SOUT lines
3. Modem input signals have not changed in more than one character time.

7.2.5.2 SPI autonomous clock gating (ACG)

All SPI controllers support ACG mode (**CCU_SPI_XX_PCLK_EN_SW=0).** ACG occurs when the SSIENR register has been written to 0.

8 Power Management

This chapter provides information on the power management and power architecture of the SoC.

Power architecture of SoC is based on the following premise:

1. There is no requirement to supply regulated voltage (1.8v or 3.3v) to platform components from SoC.
   - This dictates the max current specification of internal voltage regulator in the SoC.
2. Platform Components and SoC would operate from same battery source (eg coincell).
   - The IOs of SoC and the Analog components would be on the same rail as input battery source. Thus the electrical characteristics of SoC IOs (digital or analog) would be a function of battery rail and not regulated 1.8v/3.3v.
3. SoC has to generate a regulated 1.8v supply (DVDD) from battery input for operating its core logic.
   - There is level shifter in IOs between core voltage rail and IO rail (IOVDD). Similarly analog components such as ADC and Comparators take care of shifting from analog rail (AVDD) to core/digital rail or vice-versa.

SoC has 4 power input pins:

1. PVDD – Used by internal voltage regulator only.
2. IOVDD – Used by digital IO pads. Electrical characteristics of all external digital pins will be with respect to IOVDD.
3. AVDD – Used by Analog components such as ADC and Comparators. Electrical characteristics of all external analog pins will be with respect to AVDD.
4. DVDD – Used as SoC Core Voltage. Normal mode operating point is 1.8V. It can be fed either by internal voltage regulator's output (to be circled back in board); or from another regulated power supply from platform if internal voltage regulator is disabled. All of SoC internals will operate with DVDD.

PVDD, IOVDD, AVDD can be from the same power source with noise isolation, or can be independently supplied.

Power management of SoC is a function of how the individual component/device power state is managed. The various components in SoC that play a role in power management are Voltage Regulator, 32 MHz Oscillator, 32 kHz Oscillator, ADC, Analog Comparators, memories (SRAM, Flash), Processor Core, Peripheral controllers (digital logic) and IOs. Hence this chapter begins with a discussion on component power states and then moves on to create System power states based on component power states. The System power states are defined based on current draw requirement and latencies involved in entering or exiting a given power state. System power state is managed in Firmware/Software and not in hardware.

8.1 Component Power States

8.1.1 Voltage Regulator

Table 18. VR Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
NRML-1.8V	Normal mode. Voltage regulator (Retention Alternating Regulator) functions in switching regulator (eSR) mode and able to output 50 mA current. eSR output regulates. iLR-RET output set to high impedance. EN=H, VREGSEL=L. Change in power state incurs tSTRB of 1 usec + tROK_PROG.	90% efficiency	2 us	20 us	writing into AON_VR.VREG_SEL
RET-1.8V	Retention mode. Voltage regulator (Retention Alternating Regulator) functions in linear regulator (iLR) mode and able to output 300 uA current. iLR-RET output regulates. eSR output set to high impedance. EN=H, VREGSEL=H. Change in power state incurs tSTRB of 1 usec + tROK_PROG of latency.	1% of current load	20 us	2 us	writing into AON_VR.VREG_SEL
RET-1.35V	Retention mode. Voltage regulator (Retention Alternating Regulator) functions in linear regulator (iLR) mode and able to output 300 uA current. iLR-RET output regulates. eSR output set to high impedance. EN=H, VREGSEL=H.	1% of current load	3 us	4 us	writing into AON_VR.VSEL and AON_VR.VSTRB
OFF	HiZ mode. Voltage regulator is disabled. iLR-RET output is set to high impedance. eSR output is set to high impedance. VREF_OUT is is set to its nominal value. EN=L, VREGSEL=x.	-	-	-	Platform pulling down VR_EN input pin (use external voltage regulator)

Note: Silicon OSC minimum current is ~200uA when configured for 4MHz operation (lowest power mode) – configuring the Voltage Regulator into Low Power is only be supported when clocking off either 32.768 kHz OSC or 4 MHz Silicon oscillator mode with prescaler set for 125 kHz or lower output.

“OFF” power state is entered only when internal regulator is disabled using VREN=L input pin and 1.8v rail (DVDD input) is fed directly from platform.

Voltage regulator in eSR mode has 90% efficiency down to 0.2 Imax (10 mA, 0.2 x 50) and 70% efficiency down to 0.01 Imax (500 uA). In iLR mode, power consumption of VR is about 1% of delivered current.

The SoC power delivery is described in detail in Chapter 8.3, “Power Architecture”.

8.1.2 CPU

Table 19. CPU Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
C0	Active State Processor executing code	~5 mA @ 32MHz	3 cycles	6 cycles	Any enabled Interrupt event or reset event
C2	Stop Clock / Halt State Entered via HLT instruction Exited via Interrupt or Reset Clock to D2000 core including Local APIC and IO APIC is gated. Clock to memory subsystem (SRAM and FLASH) can also be gated if CCU_MEM_HALT_EN=1.	~50 nA	6 cycles	3 cycles	CPU executing HLT instruction provided CCU_CPU_HALT_EN=1

Processor in SoC does not support STOPCLK (for entering C2 state) but instead executing HALT instruction is used to enter C2 state wherein clock to processor core including LAPIC and IOAPIC is gated. The clock to D2000 core can be gated after min 6 core clk cycles from HALT detection (number of clock cycles is configurable and default set to 16 clock cycles). In Intel® Quark™ microcontroller D2000, processor is never power gated but only clock gated and hence processor state is always preserved. Clock is re-enabled to the processor if there is any wake event or any enabled interrupt.

8.1.3 ADC

Table 20. ADC Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON	Normal Operation. ADC is enabled for conversion with optionally enabled internal LDO. Enadc=H, enldo=H, dislvl=L. powerup time: 3-5-10 usec (min-typ-max).	1000 uA @ AVDD, 100 uA @ DVDD at 5 MSps	-	-	Writing to ADC_OP_MODE register
STBY	Standby Mode. ADC is disabled but ADC state is kept enabled by enabling internal LDO and retaining DVDD. Enadc=L, enldo=H, dislvl=L. Exit involves 1 conversion cycle.	15 uA @ AVDD, 1 uA @ DVDD	-	14 CLK cycles	Writing to ADC_OP_MODE register

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
PD	Power Down mode. ADC is disabled, calibration state is retained, DVDD is present, internal LDO is off. Enadc=L, enldo=L, dislvl=L. A calibrated conversion cycle can start immediately after internal LDO Power Up time.	1 uA @ AVDD, 1 uA @ DVDD	-	10 usec	Writing to ADC_OP_MODE register
DPD	Deep Power Down mode. ADC is disabled, calibration state is lost, DVDD can be off. Enadc=L, enldo=L, dislvl=H. Exit involves waiting for internal voltage regulator to start-up + recalibration + dummy conversion cycle. A complete calibration cycle lasts 81 clock cycles. A conversion cycle is 14 CLK cycles for 12-bit resolution.	1 uA @ AVDD, 0.5 uA @ DVDD	-	10 usec + 95 CLK cycles.	Writing to ADC_OP_MODE register

8.1.4 Comparator

Table 21. Comparator (CMP) Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON	Normal operation. CMP_PWR=H.	Fast: 20.5 uA @ AVDD, 0.82 uA @ DVDD. Slow: 2.5 uA @ AVDD, 1.4 uA @ DVDD.	-	-	Writing to CMP_PWR register
OFF	Powered down state. CMP_PWR=L.	2.7 nA	-	0.9 us	Writing to CMP_PWR register

8.1.5 32.768 kHz OSC

Table 22. 32.768 kHz OSC Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON	Normal mode. Crystal Oscillator outputs 32 kHz clock	40 nA typ, 150 nA max.	350 msec	-	Writing to OSC1_CFG0 register
OFF	Disabled mode. Output is not oscillating.	-	-	-	Writing to OSC1_CFG0 register

8.1.6 32 MHz OSC

Table 23. 32 MHz OSC Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON-SI	Silicon RC Oscillator mode. Oscillator is outputting the configured clock frequency at +/- 2% accuracy.	450 uA @ 32 MHz; 180 uA @ 4 MHz.	2 usec	-	Writing to OSC0_CFG0/1 register
ON-XTAL	Crystal Oscillator mode. Oscillator is outputting the configured clock frequency (based on crystal connected) at +/- 100 ppm accuracy.	4000 uA @ 32 MHz; 3000 uA @ 20 MHz.	2000 usec	-	Writing to OSC0_CFG0/1 register
OFF	Powered down mode.	300 nA	-	-	Writing to OSC0_CFG1.OSC0_PD register

8.1.7 SRAM

Table 24. SRAM Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
NRML	"Normal Operation Mode: this mode corresponds to read (write) operation at every clock cycle. This mode is activated by a CSN at low level at the rising edge of CK."	-	0	0	Chip Select CSN=L to SRAM asserted in a clock cycle for write or read operation

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
STBY	Power Reduction mode (stand-by). this mode corresponds to a memory that cannot perform any read (or write) operation. This mode is activated by a CSN at high level at the rising edge of CK.	-	0	0	Chip Select CSN=H to SRAM deasserted in a clock cycle for write or read operation

There is no specific control to put the SRAM into above power states.The Intel® Quark™ microcontroller D2000 has a single always-on power domain (DVDD) and hence SRAM is always powered. SRAM state is always preserved. No special retention mode is required. STBY state is entered automatically by SRAM when the chip select to SRAM is inactive in a given clock cycle.

8.1.8 Peripherals

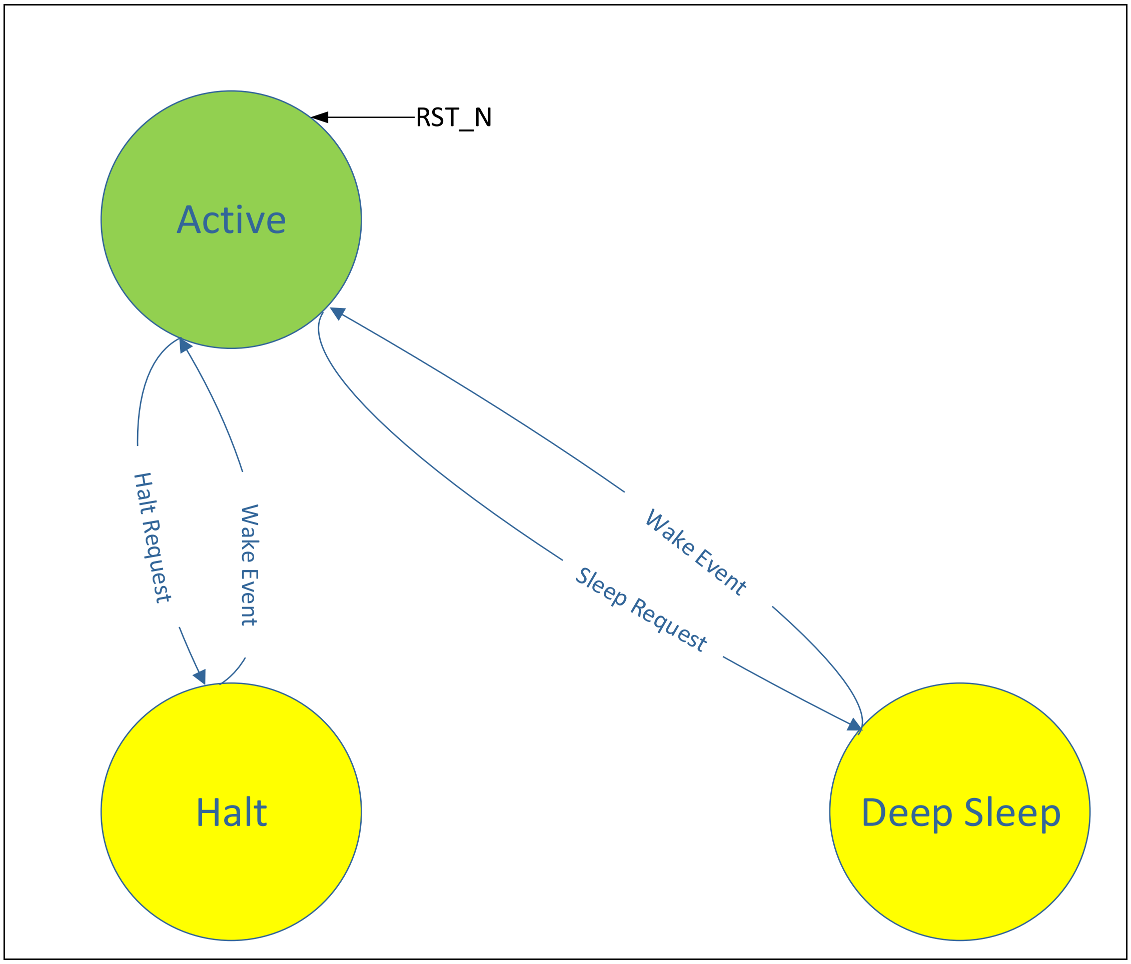
Table 25. Peripheral Power States

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON	Peripheral is enabled retaining internal state and clock to it is running	Dynamic current of logic with dependency on activity factor.	1 cycle	1 cycle	Setting respective bit in CCU_PERIPH_CLK_GATE_CTL register
STBY	Peripheral is enabled retaining internal state but clock to it is gated.	Leakage current of logic	1 cycle	1 cycle	Resetting respective bit in CCU_PERIPH_CLK_GATE_CTL register

8.2 System Power States

8.2.1 System Power State Diagram

Figure 10. System Power States



8.2.2 System Power State Definition

The Power Management states supported by the SoC are described in this section.

Table 26. SoC Power States

State	Sub State	Description
ACTIVE	RUN	Main supply rail is present and voltage regulator is in regulation in Normal mode. System clock is running from 4MHz up to 32MHz. Processor in C0. FW has full control of which peripherals to enable.
	LOW POWER COMPUTE	Main supply rail is present and voltage regulator is in regulation in Normal mode. System clock is running at less than 4MHz. Processor in C0. FW has full control of which peripherals to enable.
HALT	-	Main supply rail is present and voltage regulator is in regulation in Normal mode. FW executes the HLT instruction to enter C2. FW has full control of which peripherals to leave enabled when entering C2. Any enabled peripheral capable of generating an interrupt can trigger an exit from HALT to ACTIVE
DEEP SLEEP RTC 1.8V/1.35V	1.8V retention	Main supply rail is present and voltage regulator is in regulation in Retention mode (Linear Regulator Mode either 1.8V or 1.35V voltage output). FW has put most of the SoC components (Hybrid Oscillator, ADC, most comparators) in power down mode. FW executes the HLT instruction to enter C2. RTC oscillator is enabled. All peripherals are disabled except for AON Periodic Timer, RTC, GPIO and/or Comparator(s) which provide the wake event from DEEP SLEEP to ACTIVE.
	1.35V retention	
DEEP SLEEP NO RTC 1.8V/1.35V	1.8V retention	Main supply rail is present and voltage regulator is in regulation in Retention mode (Linear Regulator Mode either 1.8V or 1.35V voltage output). FW has put most of the SoC components (Hybrid Oscillator, ADC, most comparators) in power down mode. FW executes the HLT instruction to enter C2. RTC oscillator is also powered down. RTC alarm, AON Periodic timer are not available in this state. GPIO debouncing cannot be done in this state. GPIO edge triggered interrupt is not available as wake source as all clocks are gated off in this state. All peripherals are disabled except for GPIO level interrupt or Comparator(s) which provide the wake event from DEEP SLEEP to ACTIVE.
	1.35V retention	

8.2.3 Power and Latency Requirements

The System power states are maintained and managed in FW.

Table 27. Power and Latency Requirements

SoC Power State	Max Current		Latency		Component Power State						
	TYP 25C	TYP 105C	Entry	Exit	CPU	VR	ADC	CMP	SRAM	32 MHz	32.768 kHz
RUN	<10 mA	-	<5 usec	<2 usec	C0	NRML	Any	Any	NRML/STBY	ON	ON
LOW POWER COMPUTE	<1.2 mA	-	2 usec	<1 usec	C0 (<4MHz)	NRML/RET	Any	Any	NRML/STBY	ON/OFF	ON
HALT	<4 mA	-	6 cycles	3 cycles	C2	NRML/RET	Any	Any	STBY	ON/ OFF	ON
DEEP SLEEP RTC	<5 uA	-	<1 usec	<5 usec	C2	RET	PD/DPD	Any	STBY	OFF	ON
DEEP SLEEP NORTC	<5 uA	-	<1 usec	<5 usec	C2	RET	PD/DPD	Any	STBY	OFF	OFF

1. LOW POWER COMPUTE has to work with Hybrid Oscillator set to Silicon oscillator mode with 4MHz output to limit 32MHz C
2. "Any" state of comparator CMPH is dependent on current consumed versus number of external wakes to be enabled. Only r comparators are in ON state. 6 High performance high power comparators are in OFF state.
3. It takes ~350 msec (typ) to power up the RTC XTAL oscillator. Power saving in switching off RTC XTAL oscillator is ~150 nA.
4. Low power wait or DEEP SLEEP RTC or DEEP SLEEP NORTC corresponds to VR in retention state (capable of sourcing only 3 only in terms of which wake sources are enabled and can be decided by FW based on platform/system power usecases. If G required, the retention voltage of VR has to 1.8V typical for digital IO pads to operate.
5. Exiting DEEP SLEEP NORTC state can be with Silicon oscillator set at 4 MHz (CCU_SYS_CLK_SEL set to Hybrid oscillator at e latency is < 2 usec. RTC oscillator will lock after 350 msec (typ).
6. In DEEP SLEEP NORTC, exit is possible only with comparator based wake. GPIO Wake, AON timer wake, RTC alarm wake are
7. All entry/exit latencies are given assuming hybrid oscillator in Silicon mode and 4 MHz. Low power state Current numbers a low power comparator enabled for wake.

8.2.4 Minimum Voltage Limits (Vmin)

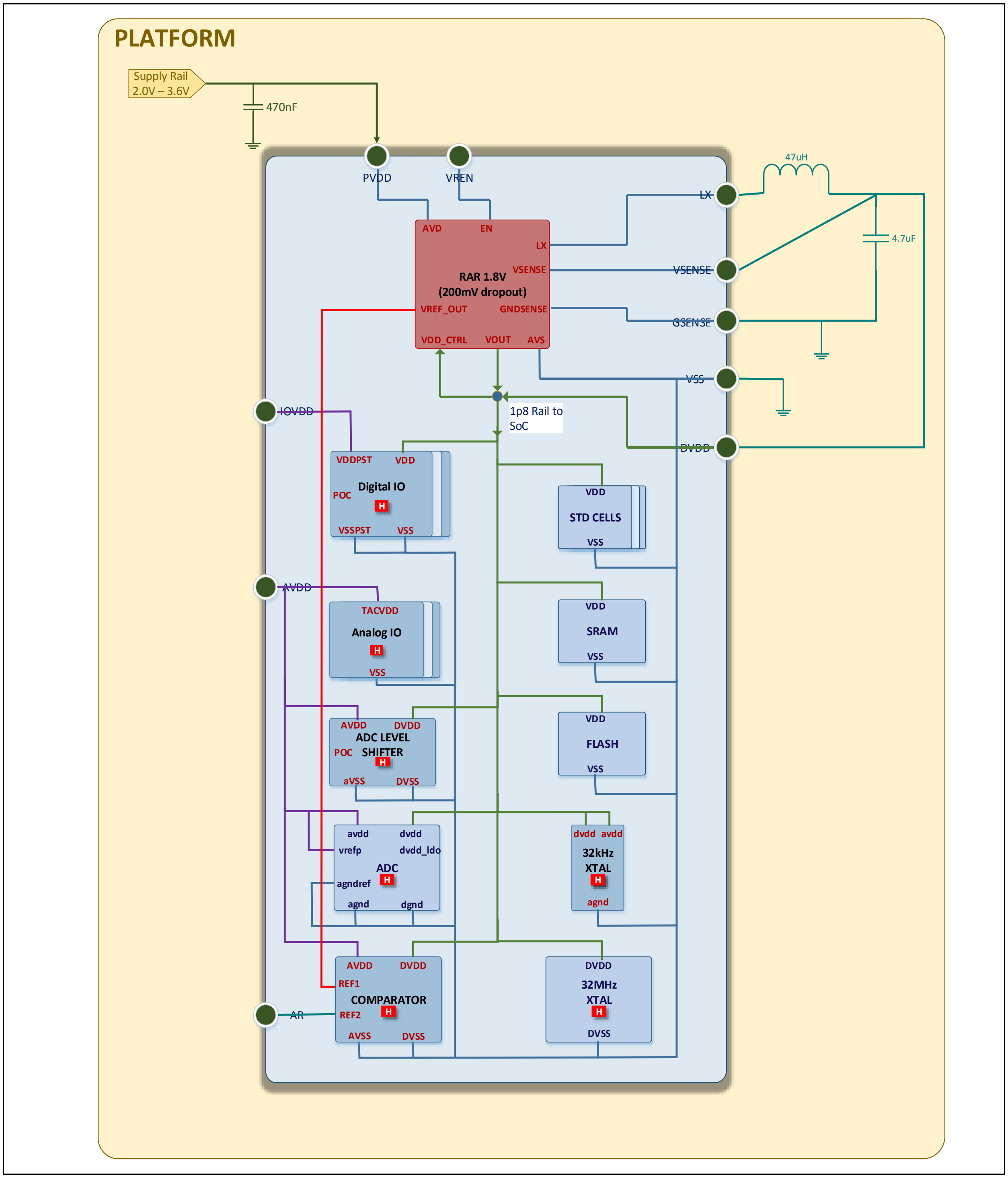
Table 28. Minimum Voltage Limits

Component	Condition	PVDD/IOVDD/AVDD		DVDD	
		Min	Max	Min	Max
RAR Voltage Regulator	Normal operation	1.62 V	3.63 V	1.08 V	3.63 V
ADC	Normal or Power Down	1.62 V	3.6 V	1.62 V	1.98 V
Comparator	Normal or Power Down	2.0 V	3.63 V	1.2 V	1.98 V
SRAM	Normal operation	-	-	1.2 V	1.98 V
Flash	Normal operation	-	-	1.2 V	1.98 V
Digital IO pads	Normal operation	-	-	1.62 V	1.98 V
Hybrid Oscillator (32 MHz)	Normal operation	-	-	1.62 V	1.98 V
	Retention mode (HYB_SET_REG1[0]=1). 4 MHz Si osc clock output	-	-	1.08 V	1.32 V
RTC Oscillator	Normal operation	1.5 V	3.63 V	1.1 V	3.63 V
Digital logic (std cells)	Normal operation	-	-	1.2 V	1.98 V

The previous table shows that DVDD has to be at 1.8V during normal operation. PVDD range is 2.0V to 3.63V limited by comparator. DVDD during retention has to be 1.8V if IOs are needed to be functional and can go below up to 1.2V (1.35V at RAR) if IOs are disabled/floating and provided Flash can work at that level. POR is only 1.8V DVDD during retention.

8.3 Power Architecture

Figure 11. Intel® Quark™ microcontroller D2000 Power Architecture



The Intel® Quark™ microcontroller D2000 power architecture is given in Figure 11 and uses a Retention Alternating Regulator (RAR). RAR works in two modes – normal mode wherein Switching Regulator is turned on sourcing 50 mA of max current and retention mode wherein linear regulator is turned on sourcing only 300 uA of max current.

The entire SoC core is under single power domain (1.8v regulated output rail from RAR) and is never power gated. Power saving is achieved by clock gating of logic and also putting the hard macros (such as ADC, comparators, oscillators, voltage regulator) in power down mode.

A tank capacitor at the output of LX is provided to ensure smooth switchover (no drop or droop) from Linear Regulator (retention mode) to Switching Regulator (normal mode) or vice-versa, provided current draw is restricted at less than 300 uA before transition.

Additionally in retention mode, RAR can supply only 300 uA of max current. Since the SoC is in single power domain fed by 1.8v regulated output from RAR, FW/SW has to ensure that enough components/devices are put into low power states such that overall current draw is less than 300 uA in LOW POWER WAIT / DEEP SLEEP states. No fail-safe scheme is implemented in the SoC.

The POC rail of the I/O ring shuts down the I/O level shifters to prevent damage while the DVDD supply comes into regulation.

Implementation may choose to separate Analog GND (RAR, ADC, Comparators) and Digital GND (Std cells, Digital IO pads, memories, Oscillators) as separate pads and ground it to VSS plane in package. Similarly VREFP and AGNDREF ports of ADC can be implemented as separate pads and bonded to AVDD and VSS respectively in package.

RST_N input uses low power comparator which has PVDD, AVDD and GND ports which are to be connected to PVDD, AVDD and VSS inputs respectively. REF1 port of RST_N comparator is connected to VREF_OUT from RAR.

Notes:

1. Current scheme assumes 40 pad QFN thus all digital and analog grounds are routed to the QFN ground pad.
2. Current scheme assumes 40 pad QFN thus several power rails are ganged together (e.g. ADC & Comparator).
3. ADC:
    a. VREFP is double bonded to AVDD
    b. AGNDREF is double boned to VSS
    c. ADC Block has an internal LDO to create a clean 1.8V rail, DVDD_LDO is provided to allow bypassing of the LDO.

8.4 Power Management Unit (PMU)

8.4.1 Internal Voltage Regulator

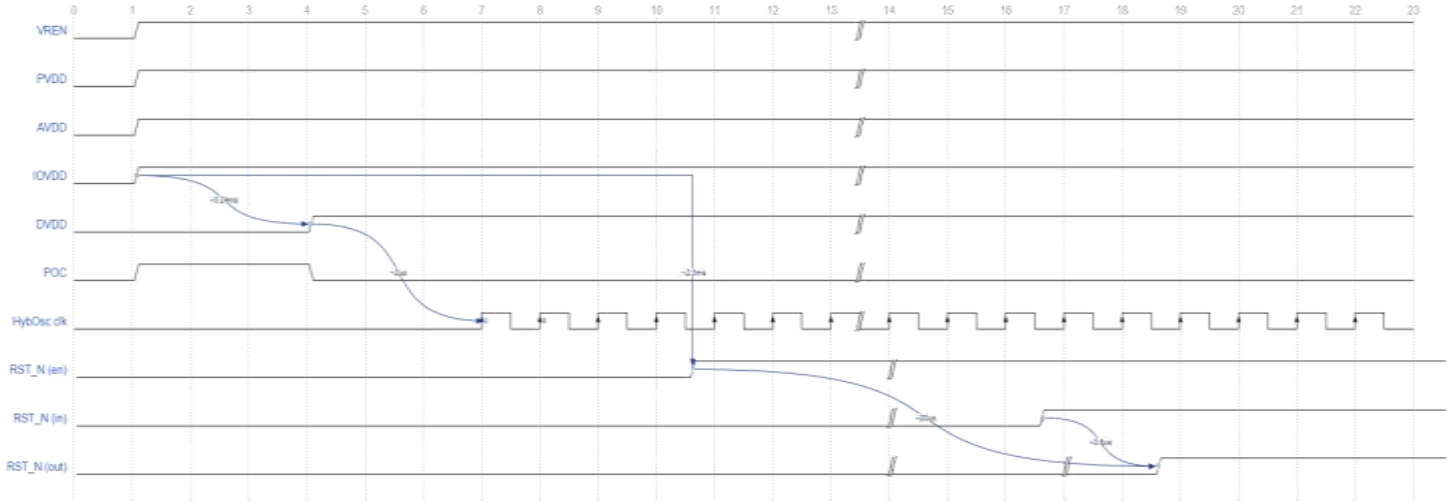
Internal voltage regulator is enabled by pulling high VREN to PVDD.

1. PVDD/AVDD/IOVDD are applied together. All these rails are from same source.
2. After 240 usec of start-up time, Voltage regulator achieves regulated 1.8V DVDD in switching voltage regulator mode.

   a. Till DVDD is stable, Power-on control (POC) of IO pads is kept asserted by an output (ROK_AVDB) from voltage regulator so that the level shifter inside IO pad between VDDPST and VDDcore is dis-engaged.

3. When DVDD is stable, Hybrid oscillator starts oscillating in Silicon RC oscillator mode outputting 4MHz +/- 40% (trimcode is not applied at this stage). HYBOSC takes 2 usec for lock time.
4. Internal voltage regulator provides a 0.95v +/- 15% internal reference voltage to the RST_N comparator within 2 msec. Till reference voltage is stable, voltage regulator sends an output to keep the RST_N comparator disabled (output = 0), thus keeping SoC under reset (internal power-on reset).
5. Once external RST_N input is deasserted and RST_N is internally enabled, SoC comes out of reset.
6. Processor is output of reset and fetches instruction at reset vector from flash. After some steps, firmware will apply the actual trimcode to HYBOSC at selected output frequency to get HYBOSC output to +/-2% accuracy.

Figure 12. D2000 SoC Boot Waveform for Internal VR Mode



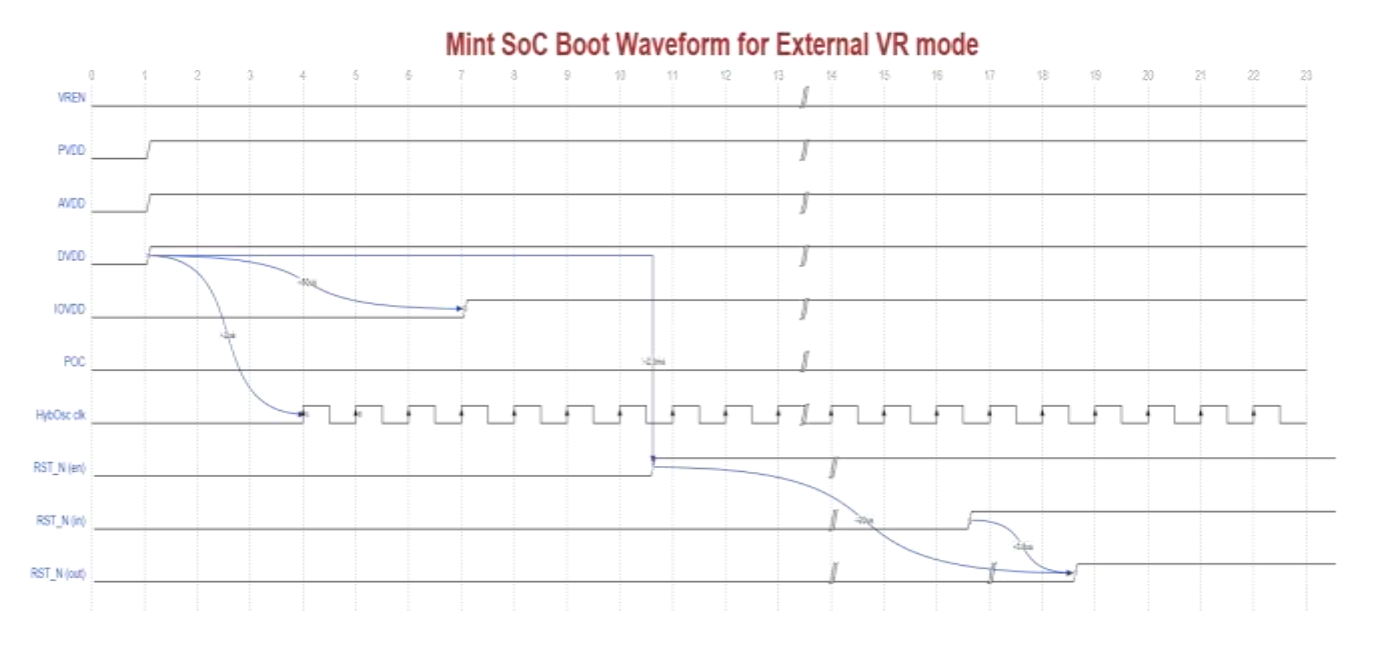
Note: Special care must be taken with the P_{VDD} power rail in the event of a power on / cycle. It is recommended to implement an Active Pull-Down Circtuit to prevent any failure in the booting up sequence. For more details refer to *"Intel® Quark™ microcontroller D2000 Platform Design Guide (Document Number: 333580-002EN)"*

8.4.2 External Voltage Regulator

Internal voltage regulator is disabled by grounding VREN to GND. In this case, DVDD voltage input is supplied by an external voltage regulator. Here DVDD has to be applied before IOVDD.

1. PVDD/AVDD/DVDD are applied together.
2. Wait for DVDD rail to ramp-up to stable regulated value.
3. As DVDD is stable, Hybrid oscillator starts oscillating in Silicon RC oscillator mode outputting 4MHz +/- 40% (trimcode is not applied at this stage). HYBOSC takes 2 usec for lock time.

4. After 50 usec minimum time, apply IOVDD rail. This is done to ensure that there is no crowbar current between VDDPST (IOVDD) and VDDcore (DVDD) in the level shifter inside digital IO Pads.
5. Internal voltage regulator provides a 0.95v +/- 15% internal reference voltage to the RST_N comparator within 2 msec. Till reference voltage is stable, voltage regulator sends an output to keep the RST_N comparator disabled (output = 0), thus keeping SoC under reset (internal power-on reset).
6. Once external RST_N input is deasserted and RST_N is internally enabled, SoC comes out of reset.
7. Processor is output of reset and fetches instruction at reset vector from flash. After some steps, firmware will apply the actual trimcode to HYBOSC at selected output frequency to get HYBOSC output to +/-2% accuracy.



9 Power Up and Reset Sequence

This chapter provides information on the following topics:

- Power Up Sequences
- Power Down Sequences
- Reset Behavior

9.1 Power Up Sequences

There are two cases of power up:

- RST_N triggered Power Up (Any state to ACTIVE state). Covers Power recycling.
- From any Low Power State to ACTIVE state (based on any of configured wake events)

Hardware (PMU) supports enabling on-die RAR Voltage Regulator, Hybrid Oscillator and RTC Oscillator during the power up sequence. Rest of the SOC components are to be enabled back by FW.

9.1.1 RST_N Triggered Transition to ACTIVE state

When RST_N is asserted following PVDD power cycle or otherwise, the following power up sequence occurs:

1. RST_N is asserted by platform. This covers the case of Power recycle as well. RST_N is kept asserted till PVDD input rail is within the operating range [2.0v-3.6v]. SoC asserts POR_RST#, COLD_RST#, WARM_RST#. RST_N is to be kept asserted for tROK_PROG (~250 usec) irrespective of whether internal voltage regulator is enabled or disabled.
    a. On-die RAR voltage regulator (VR) executes its powers up sequence whenever PVDD recycles. At other times of RST_N assertion, RAR Voltage regulator is not affected. If power cycled, RAR starts to regulate in eSR Switching Regulator mode with 1.8V voltage output.
    b. Hardware enables Hybrid Oscillator and RTC Oscillator. This is based on the default values of OSC0_CFG0/1 and OSC1_CFG registers. Hybrid oscillator is enabled in 4 MHz Silicon Oscillator mode with default TRIM code of 0. At this stage, there is no expectation to have 2% accurate oscillator output but just to have clock cycles for starting operation. Actual trimcode based on trimming is applied by FW later based on trimcode stored in Flash. Oscillators will start to oscillate once DVDD is stable above 1.62V.
2. RST_N is deasserted by platform. RST_N is expected to be asserted for tROK_PROG (250 usec) to account for internal regulator startup time. POR_RST# is removed.
3. RAR Voltage Regulator is set to eSR Switching Regulator mode with 1.8V Voltage select. This is ensured by the default values of AON_VR register.
4. PMU generates a strobe on VSEL_STROBE to RAR to put it in eSR mode at 1.8V VSEL_IN. Pulse width of VSEL_STROBE is based on PM_WAIT.VSTRB_WAIT register. At the positive edge of VSEL_STROBE, RAR deasserts ROK_BUF_VREG (if it is enabled/selected by VR_EN input).
    a. When RAR is power recycled, RAR will default to eSR 1.8V mode automatically. However at other times of RST_N, RAR may be in other mode (for example, qLR Linear

Regulator and non 1.8V). Hence this strobing is done by PMU to get it predictably back to eSR 1.8V regulation mode.

5. PMU waits for ROK_BUF_VREG from RAR to be asserted so that voltage regulator (VR) has attained regulation. In case RAR VR is bypassed by VR_EN input pin, RAR VR keeps ROK_BUF_VREG asserted always.
6. COLD_RST# and WARM_RST# are released. Clocks to D2000 Processor, Memory Subsystem (SRAM and Flash) are active.
7. Host processor D2000 starts to execute from reset vector. Later on, Firmware can enable the needed components such as ADC, comparators, peripheral subsystem to put the SoC into ACTIVE/Normal mode of operation.

9.1.2 Low Power State to Active

For those low power states not requiring voltage regulator to be put into retention (linear regulator) mode, below sequences are not required and can be handled by SW/FW by powering up needed components based on System Power State. Below sequence is applicable when voltage regulator was earlier put into retention/linear regulator mode with core output voltage set to either 1.8V or lower (say 1.35V).

When the SoC is in any of Sleep states and a wake event is triggered, the following sequence occurs:

1. An enabled wake event is triggered and latched within the SoC
2. The RTC Power Down input is asynchronously de-asserted by PMU in order to restart the 32 kHz clock
    a. Powering down the RTC oscillator is an optional step when entering sleep (say DEEPSLEEP_NORTC state).
3. Hybrid Oscillator Power Down input is asynchronously deasserted by PMU. Hybrid oscillator will come up in the same settings as it were at the time of entering low power state.
4. Based on CCU_SYS_CLK_CTL.CCU_SYS_CLK_SEL programmed by FW at the time of entering into low power state, sys_clk will take either Hybrid oscillator output or RTC clock output. This step exits based on the LOCK time of selected oscillator. If CCU_LP_CLK_CTL.CCU_EXIT_TO_HYBOSC was set to 1'b1, then PMU ensures that sys_clk takes hybrid oscillator output.
5. Once the sys_clk starts ticking, Host Processor will get the wake interrupt. D2000 starts to execute.
6. FW to restore clock settings for normal mode: OSC0_PD=0, OSC1_PD=0 or 1 (1'b1 if RTCOSC is not needed in active state), CCU_SYS_CLK_SEL = 1 (HYBOSC) to be updated to proper values.
7. FW to program HYB_OSC_PD_LATCH_EN = 1, RTC_OSC_PD_LATCH_EN=1 so that OSC0_PD and OSC1_PD values directly control the oscillators in active state.
8. FW to make WAKE_MASK[31:0] to all-ones (all wake disabled) so that any future interrupt in active power state does not interfere with wake related logic (such as powering down oscillators etc).
9. FW has to bring back RAR voltage regulator to eSR normal mode before it enables other SoC components for normal mode of operation.
    a. Set AON_VR.VREG_SEL to eSR/normal mode.
    b. Wait for 2 usec for RAR to switch back to eSR normal mode delivering up to 50 mA max current.
    c. Clear AON_VR.ROK_BUF_VREG_MASK.
10. If (SCSS AON_VR.VSEL == 1.35V) { // exiting from 1.35V core voltage mode

a. FW to set SCSS.AON_VR.VSEL = 0x10; // Set to 1.8V. This will take effect only for VSEL_STROBE. Perform read modify write along with passcode.
b. FW to do Voltage strobing in next access to AON_VR register after setting up VSEL previously.
SCSS.AON_VR.VSEL_STROBE = 1; // Bit 5. Perform read mod write along with passcode
c. Wait for 1 usec;
d. FW to reset SCSS.AON_VR.VSEL_STROBE = 0; // Bit 5. Perform read mod write along with passcode
e. Wait for 2 usec; // Wait for 1.8V to take effect
f. FW to reset SCSS.AON_VR.ROK_BUF_VREG_MASK = 0; // Perform read modify write along with passcode.
g. Wait for 1 usec; // 1 usec for DVDD to be stable at 1.8V before changing HYBOSC and Flash low voltage mode.
h. FW to move HYBOSC from low voltage retention mode to normal 1.8V mode.
SCSS.OSC0_CFG0.OSC0_HYB_SET_REG1.OSC0_CFG0[0] = 0;
i. FW to Put Flash from LVE mode to Normal Voltage mode
FlashCtrl.CTRL.LVE_MODE = 0;

11. Switch back hybrid oscillator to 32MHz frequency.
    a. FW to set CCU_SYS_CLK_CTL.CCU_SYS_CLK_DIV to needed divisor value.
    b. FW configures Hybrid Oscillator (OSC0_CFG0/1 registers) to 32 MHz Silicon oscillator mode while applying trimcode specific to 32 MHz frequency.
12. Firmware can enable the needed components such as ADC, comparators, peripheral subsystem to put the SoC into ACTIVE/Normal mode of operation including enabling PERIPH_CLK_EN to 1'b1.
    a. Note that none of the resets (say WARM_RST#) is asserted during power state transitions.
    b. since host processor subsystem is never powered down in low power state, state of the CPU core as well as other SoC components are preserved. Hence FW/OS is not expected to do a save/restore operation, thus saving time in exit latency.

9.2 Power Down Sequences

Power down sequence is totally executed by FW depending on the low power state to enter. PMU does not play any part in this sequence. Before entering low power state, the intended wake sources have to be enabled.

The possible wake sources/events that can be configured by FW/SW while entering a low power state is as below. Corresponding WAKE_MASK[31:0] and/or CCU_LP_CLK_CTL.WAKE_PROBE_MODE_MASK register bits specific to the wake source of interest has to be unmasked (set to 0) while remaining bits are to be masked (set to 1) to prevent unwanted wake.

One or more possible wake sources can be simultaneously enabled in a given low power state. RST_N assertion will automatically transition the SoC to normal/active state (4 MHz Si OSC mode).

Low Power State	Possible Wake Sources
Deep Sleep RTC state	AON Periodic Timer (AONPT), RTC Alarm, GPIO Edge/Level triggered interrupt (with or without GPIO debouncing), Comparator, CLTAP Probe mode request through JTAG , RST_N assertion.
Deep Sleep NoRTC state	GPIO Level triggered interrupt (without GPIO debouncing), Comparator interrupt, CLTAP Probe mode request through JTAG, RST_N assertion.

CLTAP Probe mode request is generated by setting CLTAP_CPU_VPREQ.assert_vpreq to 1 (TAP instruction 0x70 Bit0) through JTAG interface. In deep sleep state, ensure TCK frequency is less than 32 kHz (system clock frequency at that state divided by 4).

For setting GPIO level triggered interrupt as wake source, GPIO controller has to be programmed as below for a specific GPIO pin x of interest:

- GPIO_INTEN[x] = 1. Enable interrupt for that particular GPIO pin.
- GPIO_INTYPE_LEVEL[x] = 0; Level sensitive interrupt.
- GPIO_INT_POLARITY[x] = 1; Active-high level interrupt (0 to 1 on GPIO pin will wake; default state of this GPIO input pin shall be 0). Set this register bit to 0 if it has to be active-low level interrupt. 0 level to trigger interrupt.
- GPIO_DEBOUNCE[x] = 0. No debounce as there is no clock running.

GPIO_LS_SYNC[x] = 0. Not synchronized as there is no clock running to synchronize. (d) and (e) are important settings.

9.2.1 Active to Any Low Power State

For those low power states not requiring voltage regulator to be put into retention mode, the following sequences are not required and can be handled by SW/FW by powering down or enabling clock gating of specific components not in use based on System Power State. For example, Low Power Halt state is mainly halting processor and optionally memory system by executing HALT instruction while peripherals can be in operation.

Whenever user application / software wants to enter any low power state (having voltage regulator in retention mode) from ACTIVE state, the following sequence occurs: All steps are executed by Firmware.

1. SW/FW to ensure all interrupts are serviced and no interrupt pending.
2. SW/FW to ensure that all high power components such as ADC, high-performance comparators ([5:0]), unneeded low power comparators (18:6]), are powered down and peripheral subsystem clock gated (PERIPH_CLK_EN = 0).

3. SW/FW must enable the needed interrupt sources for wake and mask all other interrupt sources. Wake sources can be any of enabled low power comparators, any GPIO based interrupt wake, AON Periodic Timer expiry, RTC alarm interrupt. Additionally system automatically wakes up to RST_N assertion.
    a. Program WAKE_MASK.WAKE_MASK[31:0], CCU_LP_CLK_CTL.WAKE_PROBE_MODE_MASK registers identical to Interrupt Mask registers.
4. Program HYB_OSC_PD_LATCH_EN = 0, RTC_OSC_PD_LATCH_EN=0. This ensures that powering down of oscillators is delayed by hardware till core executes HALT at last step in this sequence.
5. Program CCU_LP_CLK_CTL.CCU_EXIT_TO_HYBOSC to 1'b1. This ensures that at exit, hardware will switch system clock to Hybrid oscillator clock so as to minimize exit latency by running at higher frequency than RTC clock.
6. If RTC clock is not needed during low power state (no AON Timer, RTC, GPIO interrupt), FW to program OSC1_CFG0.OSC1_PD to 1'b1. RTC Oscillator will actually get powered down only at last step when core enters HALT.
7. FW to program OSC0_CFG1.OSC0_PD = 1. Hybrid Oscillator will actually get powered down only at last step when core enters HALT.
8. FW configures Hybrid Oscillator (OSC0_CFG0/1 registers) to 4 MHz Silicon oscillator mode while applying trimcode specific to 4MHz frequency. Note that Hybrid oscillator shall not be disabled to effect this change if CCU_SYS_CLK_SEL is set to hybrid oscillator as it stops the clock to processor. This step ensures that current consumed by hybrid oscillator is reduced to ~180 uA levels as max retention mode current supply from RAR is 300 uA.
9. FW sets CCU_SYS_CLK_CTL.CCU_SYS_CLK_DIV to div-by-32 (or lower) to reduce system clock frequency to 125 kHz (or lower). This step is needed to reduce dynamic power of processor and digital logic so that overall current draw by SoC is now less than 300 uA.
10. If (retention voltage in low power state == 1.35V) { // change to 1.35V in eSR mode. This step is not needed if retention voltage is unchanged at 1.8V itself.
    a. FW to move HYBOSC to low voltage retention mode. SCSS.OSC0_CFG0.OSC0_HYB_SET_REG1.OSC0_CFG0[0] = 1;
    b. FW to Put Flash to LVE mode from Normal Voltage mode FlashCtrl.CTRL.LVE_MODE = 1;
    c. FW to set SCSS.AON_VR.VSEL = 0xB; // Set to 1.35V. This will take effect only with VSEL_STROBE. Perform read modify write along with passcode.
    d. FW to set SCSS.AON_VR.ROK_BUF_VREG_MASK = 1; // Perform read modify write along with passcode.
    e. FW to do Voltage strobing in next access to AON_VR register after setting up VSEL previously.
       SCSS.AON_VR.VSEL_STROBE = 1; // Bit 5. Perform read mod write along with passcode
    f. Wait for 1 usec;
    g. FW to reset SCSS.AON_VR.VSEL_STROBE = 0; // Bit 5. Perform read mod write along with passcode
    h. Wait for 2 usec; // Wait for 1.35V to take effect

11. FW to configure RAR Voltage regulator to operate in retention mode (Linear Regulator). This step is needed as RAR in eSR switching regulator mode is very inefficient (consumes more power) at low current loads.
    a. Set ROK_BUF_VREG_MASK as AON_VR.ROK_BUF_VREG would go low during retention mode. This ensures that logic that uses ROK_BUF_VREG output from RAR are not falsely triggered.
    b. Set AON_VR.VREG_SEL to qLR/retention mode. VSEL_IN is set to 1.8V. VSEL_STROBE strobing is not required here as voltage is not changed. Don't do any voltage programming of RAR as Retention voltage less than 1.8V is not Plan of Record, as it would make all IO pads non-functional. After 20 usec (no need to wait for this time), RAR would come up in qLR retention mode delivering 300 uA max current. External tank capacitor ensures that DVDD supply is maintained without any drops or droops during this transition.
12. If wake source is any of AON Timer, RTC, GPIO interrupt, program CCU_SYS_CLK_CTL.CCU_SYS_CLK_SEL to RTC Oscillator. This step is not needed if wake source is only comparator and/or GPIO level triggered interrupt (without GPIO debouncing).
13. SW/FW to execute HALT instruction.
    a. Once core is halted, PMU will automatically clock gate processor clock and memory subsystem clock. This reduces dynamic power consumed by processor and memory subsystems (provided CCU_LP_CLK_CTL. CPU_CPU_HALT_EN and CCU_LP_CLK_CTL.CPU_MEM_HALT_EN are enabled). And also PMU will power down Hybrid oscillator and RTC Oscillator as per respective OSC0_PD and OSC1_PD register bits.

9.2.2 Power Sequence Analog Characteristics

The following table describes the analog characteristics of the blocks used in the SoC power sequences.

Table 29. Power Sequence Analog Characteristics

Parameter	Minimum	Typical	Maximum	Units
Time taken to attain regulation (from EN=0 to 1) eSR t_{ON}		0.15		ms
Time taken to attain regulation (from EN=0 to 1) qLR t_{ON}	2	20		ms
Mode transition (VREG_SEL = 1 to 0) to VSEL_STROBE rising edge teSR_setup			2	us
Mode transition (VREG_SEL = 0 to 1) to VSEL_STROBE rising edge t_{SETUP}	500			ns
VSEL_STROBE pulse width t_{STRB}	1			us

9.2.3 Handling Power Failures

Power failure can occur if main power or battery is removed or brownout occurs.

On-die voltage regulator requires minimum 2.0V at PVDD to ensure 1.8V DVDD for normal operation.

The SoC does not provide any brownout detection or indicator and relies on external platform agents to handle brownout and to assert RST_N input pin.

9.3 Reset Behavior

The SoC supports three types of reset:

- Power On Reset
- Cold Reset
- Warm Reset

9.3.1 Power On Reset

The SoC provides on-die circuitry to provide a power on reset when main power is applied. The power on reset is asserted when the SoC is powering up and is released when VCC_AON_1P8 has crossed a given threshold for a certain length of time.

The only mechanism to trigger a power on reset is to remove and then re-apply main power.

The only indication to the SoC that power recycling occurred is the RST_N input pin. Whenever there is power recycle, RST_N is expected to be asserted and then de-asserted once input power rail PVDD is within the operating voltage range [2.0V to 3.6V]. In case of a power recycle, RST_N has to be kept asserted till DVDD core voltage is also stable (internal voltage regulator has attained regulation, tROK_PROG < 100 usec). Hence RST_N is taken as proxy of power-on reset

RST_N could be asserted at other times as well to trigger a complete SoC reset. Only a PVDD power recycle will restart the on-die voltage regulator. RST_N will reset the entire SoC including System Control Subsystem (SCSS) and registers.

When RST_N is triggered, the following sequence occurs:

1. A RST_N event is detected.
2. SCSS asserts POR_RST#, COLD_RST# and WARM_RST#.
3. Waits till RST_N is deasserted. Then deasserts POR_RST# used by SCSS logic (PMU, CCU).
4. RAR Voltage Regulator is set to eSR Switching Regulator mode. Ensured by default value of configuration register (AON_VR.VREG_SEL).
5. The RTC Power Down input is de-asserted. Hybrid Oscillator is enabled in Silicon Oscillator mode at 4MHz Frequency Select. This is done by default values of corresponding SCSS registers (OSC0_CFG0/1, OSC1_CFG0) which are reset at cold reset.
    a. This is to ensure that if the cold reset was triggered while in a Sleeping state with the RTC disabled, the 32 kHz clock and Hybrid Si Oscillator 4MHz get restarted.
    b. The mux select config register (CCU_SYS_CLK_SEL) to choose system clock is automatically set to select Hybrid Si Oscillator [Intel® Quark™ microcontroller D2000 can work without RTC clock].
    c. Once the clock is running, the PMU accepts the cold reset request and asserts both COLD_RST# and WARM_RST#.
6. PMU generates a strobe on VSEL_STROBE to RAR to put it in eSR mode at 1.8V VSEL_IN. Pulse width of VSEL_STROBE is based on PM_WAIT.VSTRB_WAIT register. At the positive edge of VSEL_STROBE, RAR deasserts ROK_BUF_VREG (if it is enabled/selected by VR_EN input).
    a. When RAR is power recycled, RAR will default to eSR 1.8V mode automatically. However at other times of RST_N, RAR may be in other mode (for example, qLR Linear Regulator and non 1.8V). Hence this strobing is done by PMU to get it predictably back to eSR 1.8V regulation mode.
7. PMU waits for ROK_BUF_VREG from RAR to be asserted so that voltage regulator (VR) has attained regulation. In case RAR VR is bypassed by VR_EN input pin, RAR VR keeps ROK_BUF_VREG asserted always.
8. COLD_RST# and WARM_RST# are released.

9.3.2 Cold Reset

A cold reset will trigger a power cycle of the Host domain (Processor Subsystem, Memory Subsystem, Peripheral Subsystem and Fabric) and trigger a reset of registers both in the Host and AON (SCSS) domains. There is no reset cycling of most of AON domain (SCSS) logic and also certain SCSS registers due to a cold reset.

Table 30. Cold Reset Triggers

Trigger	Description
Software writes 1 to RSTC.COLD	Software Initiated via Reset Control Register

When a cold reset is triggered, the following sequence occurs:

1. A cold reset event is detected.
2. SCSS asserts COLD_RST# and WARM_RST#. Note that POR_RST# is not asserted; RAR Voltage Regulator is not affected and continues to operate in same mode as before.
3. Wait for cold reset triggers to get cleared.
    a. Note that RSTC.COLD register bit gets cleared at COLD_RST#.
4. COLD_RST# and WARM_RST# are released.

9.3.3 Warm Reset

A warm reset will trigger a reset of all Host domain logic and all non-sticky registers. There is no power cycling of the Host or AON domains due to a warm reset. Intel® Quark™ microcontroller D2000 is single always-on power domain for the entire design (DVDD).

Table 31. Warm Reset Triggers

Trigger	Description
Software writes 1 to RSTC.WARM	Software Initiated via Reset Control Register
Watchdog Expires	-
Host Halt Interrupt (Redirected to warm reset)	This is achieved by unmasking HOST_HALT_MASK register bit of a given interrupt source along with P_STS.HALT_INT_REDIR = 0. Restriction of interrupt sources is given in Note 1 below.
lmt.shutdown	Host processor shutdown. This triggers warm reset instead of CPU-only reset.

When a warm reset is triggered, the following sequence occurs:

1. A warm reset event is detected.
2. SCSS asserts WARM_RST#. This resets host domain including processor core.
3. Wait for Warm Reset triggers to get cleared.
    a. Note that RSTC.WARM register bit gets cleared at WARM_RST#. Similarly watchdog timer, processor and other interrupt generation blocks are reset by warm reset.
4. WARM_RST# is released.

Note 1: Following interrupt sources are not to be redirected to trigger warm reset as they will not get cleared due to warm reset, leading to SoC permanently under warm reset. Only a power recycle or RST_N recycle will recover this condition.

1. RTC Interrupt (INT_RTC_HOST_HALT_MASK register shall remain masked permanently – default value)
2. Comparator Interrupt (INT_COMPARATORS_HOST_HALT_MASK[18:0] register shall remain masked permanently – default value)
3. AON Timer Interrupt (INT_AON_TIMER_HOST_HALT_MASK register shall remain masked permanently – default value)

10 Thermal Management

10.1 Overview

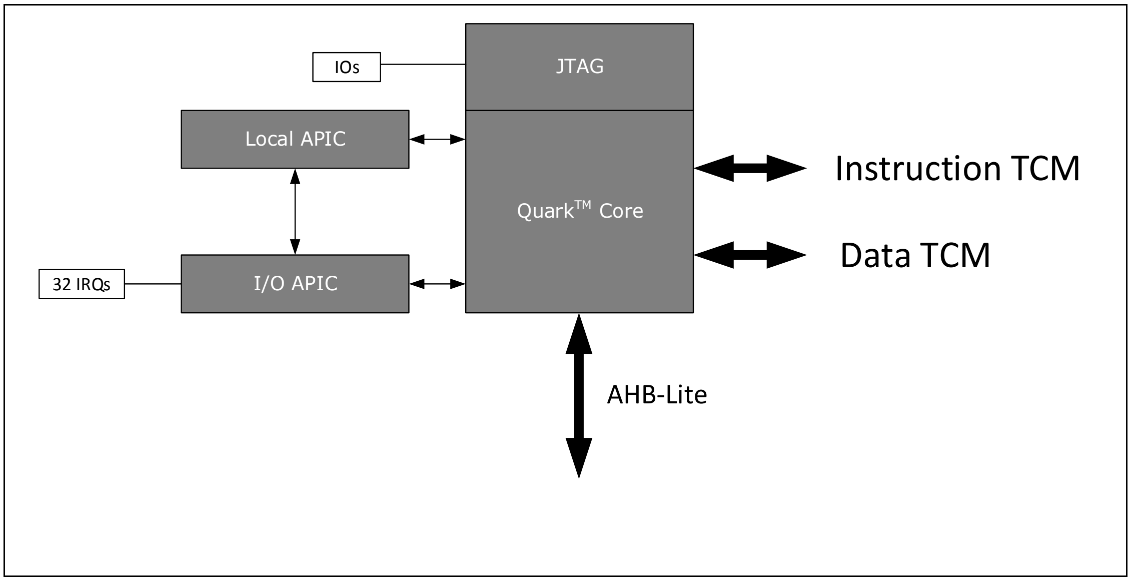
The Intel® Quark™ microcontroller D2000 SoC does not contain an integrated thermal sensor.

Ambient temperature = -40°c to +85°c.

11 Processor Core

The SoC provides a single core x86 processor with separate and independent Tightly Coupled Memory (TCM) Interfaces for Instruction and Data.

Figure 13. Processor Core



11.1 Features

- Single Processor Core
- Single Instruction 5-stage pipeline
- 32-bit Processor with 32-bit Data Bus
- Native 32b AHB-Lite Interface
- 64b Data TCM Interface to Internal System SRAM
  - Data Transfers for addresses matching the Internal System SRAM range will appear on the Data TCM Interface and transfers to address outside this range will appear on the AHB-Lite Interface
- 64b Instruction TCM Interface to Internal System Non-Volatile-Memory (NVM)
  - Instruction Fetches for addresses matching the Internal NVM range will appear on the Instruction TCM Interface and transfers to address outside this range will appear on the AHB-Lite Interface
- Support for IA 32-bit with Pentium x86 ISA compatibility
  - Reset Vector of 0x0000_0150
  - Little Endian
- Support for CPUID Instruction
- Support for long NOP Instruction

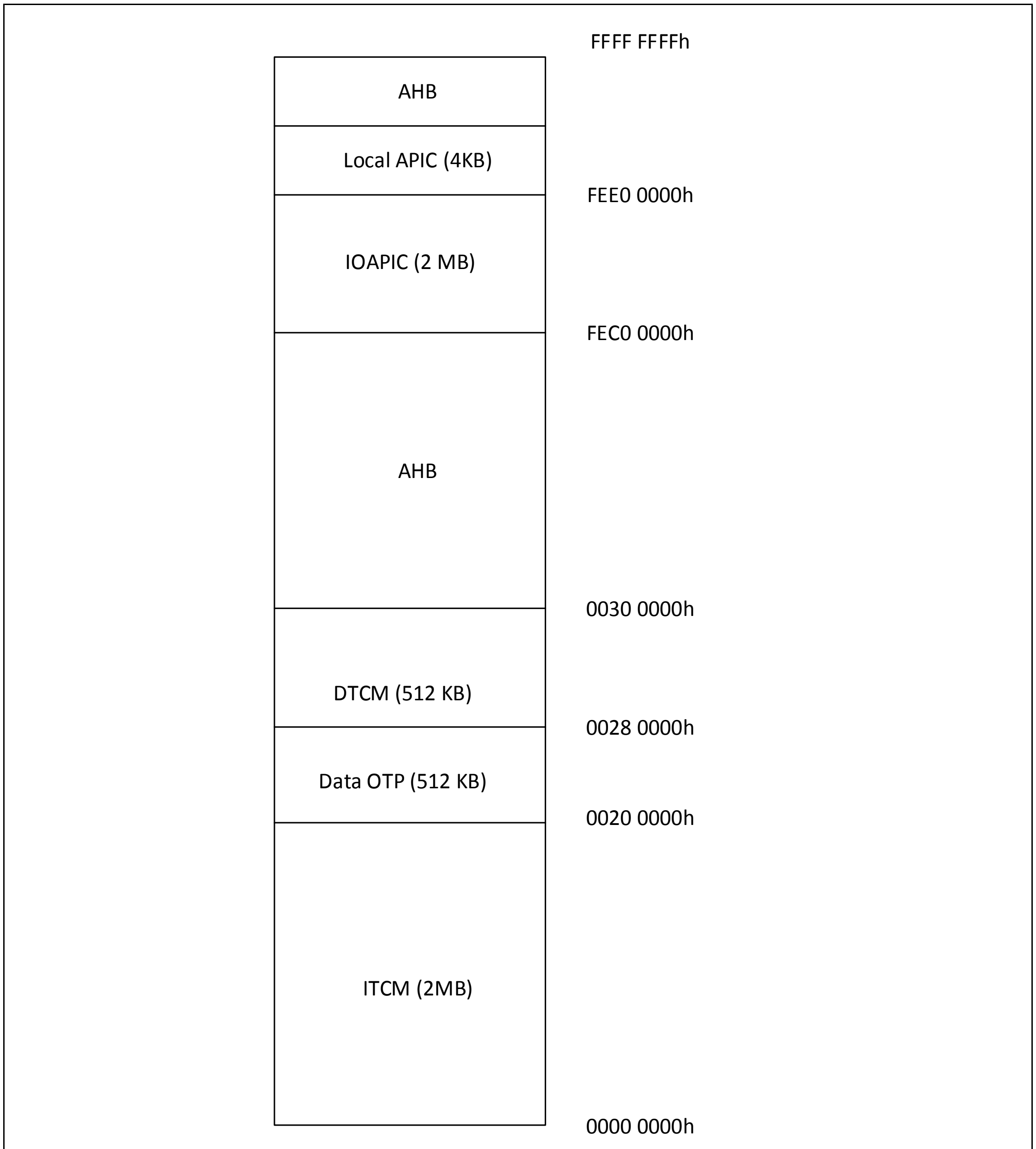
- Time Stamp Counter (TSC) accessed with the RDTSC instruction
- Support for Paging included although not required for the Intel® Quark™ microcontroller D2000 use case
  - 2 Entry Instruction TLB (Translation Look-aside Buffer)
  - 2 Entry Data TLB (Translation Look-aside Buffer)
- Single cycle 32bx32b → 32b Multiplier (IMUL Instruction)
- Integrated Intel® Quark™ microcontroller D2000 Interrupt Controller (MVIC) with support for 32 IRQs – some may be unused in Intel® Quark™ microcontroller D2000.
- Supports C0 and C1 Processor Power States
  - Supports Interrupt as Wake Event from C1
  - Both Time Stamp Counter and LVT Timer run in C1 State
  - STOP CLOCK feature is not supported
  - Intel® Quark™ microcontroller D2000 implements C2 capability of processor by clock gating processor upon detecting HALT instruction
  - Time Stamp Counter and LVT Timer do not run when Intel® Quark™ microcontroller D2000 clock gates processor

Note: The processor does not provide an x87 Floating Point Unit (FPU) and does not support x87 FPU instructions.

11.2 Processor Memory Map

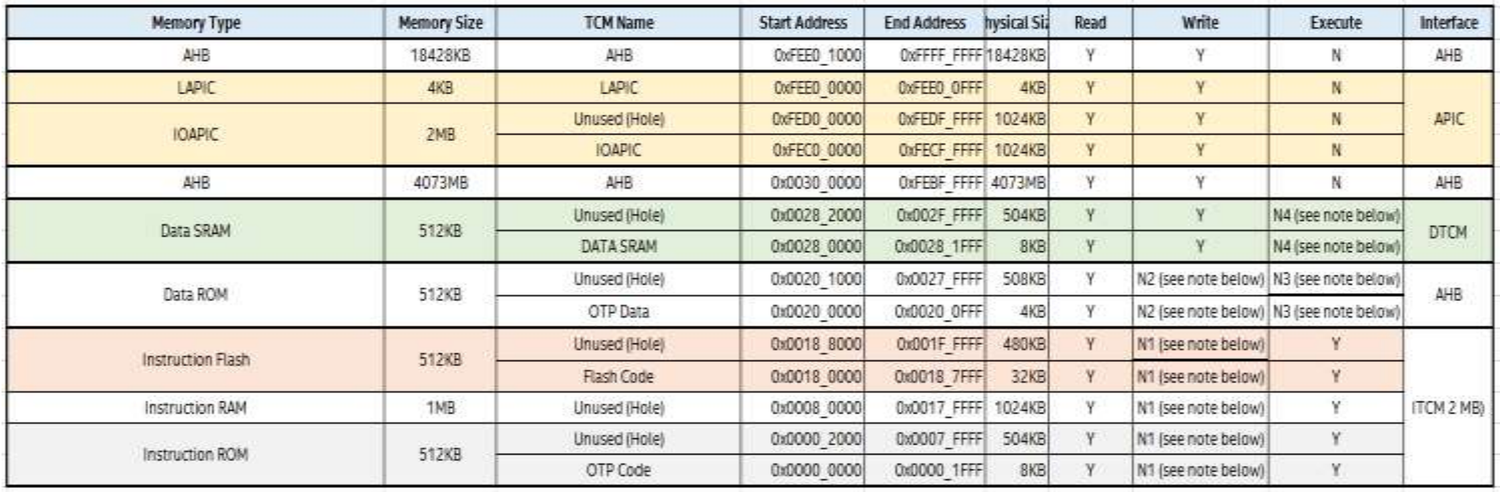
The processor memory map for the Intel® Quark™ microcontroller D2000 SoC shall cater for the planned subsequent derivative SoCs which are likely to include variations in the amount of NVM and SRAM included in the SoC. Figure 14 shows the generic processor memory map that will be applicable to Intel® Quark™ microcontroller D2000 and its derivatives.

Figure 14. Generic Intel® Quark™ microcontroller D2000 Processor Memory Map



The Reset Vector for the processor resides at 0000_0150h.

The CPU address map for Intel® Quark™ microcontroller D2000 is as follows:



Memory Type	Memory Size	TCM Name	Start Address	End Address	hysical Si	Read	Write	Execute	Interface
AHB	18428KB	AHB	0xFEE0_1000	0xFFFF_FFFF	18428KB	Y	Y	N	AHB
LAPIC	4KB	LAPIC	0xFEE0_0000	0xFEE0_0FFF	4KB	Y	Y	N	APIC
IOAPIC	2MB	Unused (Hole)	0xFED0_0000	0xFEDF_FFFF	1024KB	Y	Y	N	
		IOAPIC	0xFEC0_0000	0xFECF_FFFF	1024KB	Y	Y	N	
AHB	4073MB	AHB	0x0030_0000	0xFEBF_FFFF	4073MB	Y	Y	N	AHB
Data SRAM	512KB	Unused (Hole)	0x0028_2000	0x002F_FFFF	504KB	Y	Y	N4 (see note below)	DTCM
		DATA SRAM	0x0028_0000	0x0028_1FFF	8KB	Y	Y	N4 (see note below)	
Data ROM	512KB	Unused (Hole)	0x0020_1000	0x0027_FFFF	508KB	Y	N2 (see note below)	N3 (see note below)	AHB
		OTP Data	0x0020_0000	0x0020_0FFF	4KB	Y	N2 (see note below)	N3 (see note below)	
Instruction Flash	512KB	Unused (Hole)	0x0018_8000	0x001F_FFFF	480KB	Y	N1 (see note below)	Y	ITCM 2 MB)
		Flash Code	0x0018_0000	0x0018_7FFF	32KB	Y	N1 (see note below)	Y	
Instruction RAM	1MB	Unused (Hole)	0x0008_0000	0x0017_FFFF	1024KB	Y	N1 (see note below)	Y	
Instruction ROM	512KB	Unused (Hole)	0x0000_2000	0x0007_FFFF	504KB	Y	N1 (see note below)	Y	
		OTP Code	0x0000_0000	0x0000_1FFF	8KB	Y	N1 (see note below)	Y	

Notes:

- Reset Vector from CPU is mapped to 0x150 and falls into OTP Code region.
- N1: D2000 routes memory writes to Instruction regions towards AHB. These writes are dropped by SoC's Memory subsystem.
- N2: D2000 routes writes to Data ROM region towards AHB. These writes are dropped by SoC's Memory subsystem.
- N3: These requests are treated, by SoC, in the same manner (including Access Control) as normal memory read requests from D2000.
- N4: D2000 routes such requests towards AHB. These requests are treated, by SoC, in the same manner (including Access Control) as normal memory read requests from D2000.
- Note that OTP Data (Data ROM Memory Type in 1st column of above table) resides on AHB interface – from D2000 perspective.
- Code accesses, if any, to DATA SRAM or DATA ROM regions (in 1st column of above table) are routed to AHB by D2000. Similarly, data accesses, if any, to Instruction Flash or Instruction RAM or Instruction ROM (in 1st column of above table) are routed to AHB by D2000.
- Self-modifying code is not supported in Intel® Quark™ microcontroller D2000.
  - However, if it happens, D2000will route writes to Instruction Flash region towards AHB. These writes are dropped by SoC's Memory subsystem. A following read from D2000will appear on ITCM returning incorrect/previous value in Flash.
- Intel® Quark™ microcontroller D2000 always completes a request on ITCM, DTCM and AHB interfaces when address is out of bounds or there is an access violation.
- System addresses 0xFFFF_FFF0 and 0xFFFF_FFF8 are a special case by D2000and get aliased to reset vector.

11.3 Main Fabric Bus Cycle Processing

The D2000 CPU supports the following AHB-lite cycles:

- Code Read
- Memory Read (Data)
- Memory Write (Data)

The following sections describe the behavior of the SoC for all these supported types.

11.3.1.1 Code Reads

Code Reads that fall within the I-TCM memory address range will be forwarded to the I-TCM interface. Code Reads outside of the ITCM range will be forwarded by default to the AHB-lite fabric, this includes code reads to the DTCM range. Immediate data in code fetches are also routed to AHB.

Access latency, as seen by the processor, to both Flash Code and OTP Code regions is the same.

Address on ITCM interface is 19b DW address. Hence, total ITCM address space is 2MB of space. Lowest address bit is always driven to 0 by the processor since ITCM is 64b wide.

Address, issued by D2000, on ITCM is relative address. It starts from offset 0 with respect to base address of 0x0.

All accesses on ITCM are 8B address aligned and 64b access. There is no burst and no byte-enables. Hence, processor performs 2 read accesses for every 16B cacheline.

Processor cannot write to ITCM interface (either as probe mode or in any other way).

Protocol on ITCM allows for variable wait-state from Flash.

Processor routes all probe mode accesses towards AHB-Lite.

Attribute HPROT[0] = 0 on AHB-Lite interface indicates Code reads while HPROT[0] = 1 indicates Data accesses.

There is no burst support on AHB interface.

11.3.1.2 Memory Reads and Memory Writes

Memory accesses (data), both read and writes that fall within the D-TCM memory range will be forwarded to the D-TCM interface. Memory accesses outside of the D-TCM range will be forward to the AHB-lite fabric.

Memory accesses (data) are not allowed to access the I-TCM. Accesses to that region will be forward to the AHB-lite fabric.

Processor only issues a single 32b request. All accesses on DTCM are 8B address aligned and 64b access. Byte Enables indicate which 32b on 64b DTCM is valid – for both reads and writes.

Address on DTCM interface is 18b DW address. Hence, total DTCM address space is 1MB. Lowest 3 address bits are always driven to 0's by the processor.

Address, issued by D2000, on DTCM, is relative address and starts from offset 0 with respect to base address of 0x0028_0000. Only 512KB [0x0028_0000 to 0x002F_FFFF] is mapped to DTCM on D2000.

D2000 is an in-order machine with a single instruction in flight. AHB response in the fabric for a memory write makes the write on AHB-Lite interface posted. Since there is no L1 cache for processor, all transactions are uncached and hence serialized by processor.

Processor routes all probe mode accesses towards AHB-Lite.

11.3.1.3 IO Reads and IO Writes

IO reads and writes are under SW control and SW must not issue them. These requests are aliased into Memory address space on AHB-Lite interface.

11.3.1.4 Interrupt Acknowledge

Interrupt Acknowledge cycles are expected to be completed by the integrated MVIC. There is no external interrupt controller connected to the AHB fabric that could provide the interrupt vector information.

11.3.1.5 Special Cycles

The D2000 processor provides special bus cycles to indicate that certain instructions have been executed or certain conditions have occurred internally. This section describes how the Intel® Quark™ microcontroller D2000 SoC handles each of the special cycles.

11.3.1.5.1 Write-Back/Sync Special Cycle

The Writeback Special Cycle is generated by an x86 processor when a WBINVD instruction is executed.

As the processor does not have an L1 cache, the WBINVD instruction is not required to be executed and the Writeback Special cycle is not expected to appear on the main fabric bus interface.

If the code contains a WBINVD, the processor’s behavior shall be to be to treat it as a NOP.

11.3.1.5.2 Flush Ack Special Cycles

First Flush Acknowledge and Second Flush Acknowledge Special Cycles are generated by D2000-class processor to indicate the completion of a cache flush in response to the FLUSH# pin being asserted.

As the processor does not have a L1 cache, the FLUSH# pin will not used and Flush Acknowledge Special Cycles are not expected to appear on the AHB-Lite interface.

If the processor generates a Flush Acknowledge Special cycle, it will be internally acknowledged to allow the processor to make forward progress but it will not appear on the AHB fabric or on any other external interface

11.3.1.5.3 Flush Special Cycle

The Flush Special Cycle is generated by an x86 processor when an INVD instruction is executed.

As the processor does not have an L1 cache, the INVD instruction is not required to be executed and the Flush Special cycle is not expected to appear on the main fabric bus.

If the code contains an INVD, the processor's behavior shall be to be to treat this instruction as a NOP.

11.3.1.5.4 Shutdown Special Cycle

The Shutdown Special Cycle is generated by D2000 when a triple fault occurs. The special cycle indicates that the processor has ceased program execution and is in the shutdown state. The processor must be reset in order to exit the shutdown state.

If the processor generates a Special Cycle, it will be internally acknowledged and an output on the external interface will be asserted. This signal can be used by an external system agent to issue a reset to the processor.

In response to Shutdown, Intel® Quark™ microcontroller D2000 performs a warm reset of SoC and cause of reset is logged in a sticky register.

11.3.1.5.5 Halt Special Cycle

The Halt Special Cycle is generated by an x86 processor when a HLT instruction is executed.

If the processor generates a Halt Cycle, it will be internally acknowledged and an output on the external interface will be asserted. This signal can be used by an external system agent to issue take an action for power management or to track the state of the processor.

When Intel® Quark™ microcontroller D2000 detects Halt cycle, it waits for a programmable number of clocks (>6 clocks) before clock-gating the processor. Any break event (interrupt or Probe Mode activity) restarts the clock. Clock is not gated if any break event is pending. D2000 JTAG activity does not ungate clock.

11.3.1.5.6 Stop Grant Acknowledge Special Cycle

The Stop Grant Acknowledge Special Cycle is generated by an x86 processor when the processor enter the Stop Grant state in response to STPCLK# being asserted.

If the processor generates a Stop Grant Acknowledge Special Cycle, it will be internally acknowledged and an output on the external interface will be asserted. This signal can be used by an external system agent to issue take an action for power management or to track the state of the processor.

Intel® Quark™ microcontroller D2000 does not assert STPCLK# and hence Stop Grant Acknowledge cycle is not generated by processor.

11.3.1.6 MSI

The SoC AHB fabric will not send MSIs to the processor so an external interface for MSIs is not required.

MSIs may be exchanged between the components within integrated MVIC. However, these MSIs remain internal to the processor sub-system and not no appear on the AHB fabric.

The SoC does not assert NMI pin of D2000.

11.3.1.7 End of Interrupt

The processor subsystem provides an integrated MVIC. There are no other interrupt controllers in the SoC. As a result, there is no need to signal EOI information to any agent connected on the AHB fabric.

EOI information may be exchanged between the components within MVIC.

11.3.2 Mapping FSB to AHB

The processor core is a master on the internal SoC AHB fabric and a gasket to convert from FSB protocol to AHB is provided. The operation of the gasket and the interface to the AHB fabric is transparent to software.

11.3.2.1 Byte Enables

For read accesses on AHB-Lite interface, D2000 always asserts all byte enables.

D2000 does not issue burst reads or writes on AHB-Lite interface. For single writes, byte enable handling is described in the following paragraphs.

D2000 allows all combination of byte enables for 32-bit accesses provided that there is at least one enabled byte and that the enabled bytes are contiguous. This gives 10 valid 4-bit combinations for the byte enables, allowing 8-, 16-, 24- and 32-bit transfers.

AHB only allows 8-, 16- and 32-bit transfers in a single beat as specified by HSIZE. There is no support for 24-bit transfers.

In addition, the AMBA specification states that all transfers within a burst must be aligned to the address boundary equal to the size of the transfer as specified by HSIZE. This means that 16-bit transfers must start at 16b address boundary. In certain cases (unaligned 16b transaction or 24b transactions), processor splits them into 2 independent transactions on AHB Lite interface. AMBA specification also requires all transfers be aligned to address boundary equal to the size of the transfer.

As described in Table 32, only 7 combinations are generated by processor on AHB-Lite interface.

Table 32. Mapping D2000 Bytes Enables to AHB

D2000			AHB	
Byte Enables	Transfer Size	ADDR[1:0]	HSIZE[2:0]	HADDR[1:0]
0000b	No Valid Bytes	-	-	-
0001b	8-bit	00b	000b	00b
0010b	8-bit	01b	000b	01b
0011b	16-bit	00b	001b	00b
0100b	8-bit	10b	000b	10b
0101b	Non-Contiguous Bytes (Note1)	-	-	-
0110b	16-bit	01b	Not Supported	Not Supported
0111b	24-bit	00b	Not Supported	Not Supported

D2000			AHB	
Byte Enables	**Transfer Size**	**ADDR[1:0]**	**HSIZE[2:0]**	**HADDR[1:0]**
1000b	8-bit	11b	000b	11b
1001b	Non-Contiguous Bytes	-	-	-
1010b	Non-Contiguous Bytes	-	-	-
1011b	Non-Contiguous Bytes	-	-	-
1100b	16-bit	10b	001b	10b
1101b	Non-Contiguous Bytes	-	-	-
1110b	24-bit	01b	Not Supported	Not Supported
1111b	32-bit	00b	010b	00b

Note1: Processor does not issue transactions with Non-Contiguous Bytes.

AHB Fabric returns all 0's as data if address is out of bound. On DTCM, when address does not fall into SRAM region, SRAM controller returns data that is programmable via a register. Similarly, on ITCM, when address does not fall into Flash Code or OTP Code regions, Flash controller returns data that is programmable via a register.

11.4 Intel® Quark™ microcontroller D2000 Interrupt Controller (MVIC)

The Intel® Quark™ microcontroller D2000 programmable interrupt controller is based on an extension of the interrupt controller in Intel® Quark™ microcontroller D1000. The MVIC (Intel® Quark™ microcontroller D2000 Interrupt Controller) is configured by default to support 32 external interrupt lines. Unlike the traditional IA LAPIC/IOAPIC, the interrupt vectors in MVIC are fixed and not programmable. In addition, the priorities of these interrupt lines are also fixed. The interrupt vectors corresponding to the 32 interrupt lines respectively are shown in Table 33.

Table 33. MVIC Interrupt Vector Assignment

Interrupt Line	Vector
0	0x20
1	0x21
2	0x22
3	0x23
4	0x24
5	0x25
6	0x26
7	0x27
8	0x28
9	0x29
10	0x2a
11	0x2b
12	0x2c
13	0x2d
14	0x2e
15	0x2f
16	0x30
17	0x31
18	0x32
19	0x33
20	0x34
21	0x35
22	0x36
23	0x37
24	0x38
25	0x39
26	0x3a
27	0x3b
28	0x3c
29	0x3d
30	0x3e
31	0x3f

The higher the vector number, the higher the priority of the interrupt. Higher priority interrupts preempt lower priority interrupts. Lower priority interrupts do not preempt higher priority interrupts. The MVIC holds the lower priority interrupts pending until the interrupt service routine for the higher priority interrupt writes to the End of Interrupt (EOI) register. After an EOI write, the MVIC asserts the next highest pending interrupt.

11.4.1 MVIC Registers

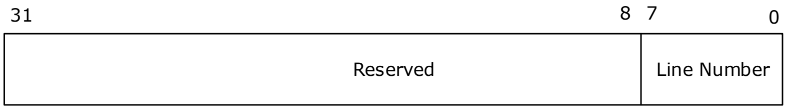
Table 34 enumerates all the programmable registers in the MVIC:

Table 34. MVIC registers

Memory Mapped Address	Register Name	Access	Description
FEE00080h	TPR	R/W	Task Priority Register
FEE000A0h	PPR	RO	Process Priority Register
FEE000B0h	EOI	WO	End-of-Interrupt Register
FEE000F0h	SIVR	R/W	Spurious Interrupt Vector Register
FEE00110h	ISR	RO	In-Service Register
FEE00210h	IRR	RO	Interrupt Request Register
FEE00320h	LVTTIMER	R/W	Local Vector Table Timer Register
FEE00380h	ICR	R/W	Timer Initial Count Register
FEE00390h	CCR	RO	Timer Current Count Register

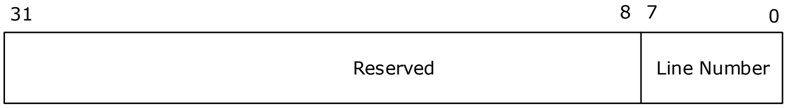
11.4.1.1 TPR

- SW writes to this register with a line number to set a priority threshold. The MVIC will not deliver unmasked interrupts with line number lower than the TPR value.
- Since the vectors are fixed, the TPR is programmed with the corresponding interrupt line number (0 to 31). Software should NOT program the vector into the TPR register. The line number to vector mapping is done internal to the MVIC.
- If SW programs the TPR as 32, then all un-masked interrupts will not be delivered.
- Register Description :



11.4.1.2 PPR

The MVIC sets the PPR to either the highest priority pending interrupt in the ISR or the current task priority, whichever is higher.



11.4.1.3 EOI

The EOI is set when CPU initiates a write to address FEE000B0h. Upon receipt of the EOI write, the MVIC clears the highest-priority ISR bit, which corresponds to the interrupt that was just serviced. The MVIC ignores the actual value written to the EOI Register.

31	0
0/Ignored	

11.4.1.4 SIVR

SW writes the vector used for spurious interrupts to the SIVR

31	8	7	0
Reserved		Vector	

11.4.1.5 ISR

This register tracks interrupts that have already requested service to the core but have not yet been acknowledged by SW. The MVIC set the bit in ISR (In-Service Register) after the core recognizers the corresponding interrupt. The bit in the ISR is cleared when SW writes to the EOI register. Bit N corresponds to the interrupt request N for interrupt vectors 32 to 63.

31	30		1	0
63	62	-----------------------------	33	32

11.4.1.6 IRR

This register contains the active interrupt requests that have been accepted, but not yet dispatched to the core for servicing. When the MVIC accepts an interrupt, it sets the bit in the IRR that corresponds to the vector of the accepted interrupt. When the core is ready to handle the next interrupt, it will sent an INTA cycle and the MVIC clears the highest priority IRR (Interrupt Request Register) bit that is set and sets the corresponding ISR bit. Note that if the interrupt line for the IRR is not cleared, then the IRR bit will NOT be cleared when the INTA is sent by the CPU.

31	30		1	0
63	62	------------------------------	33	32

11.4.1.7 LVTTIMER

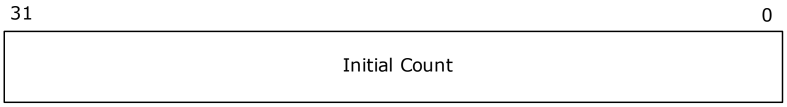
- Is used to inject an interrupt when the timer inside the MVIC expires.
- In the current implementation, lines 0 to 15 can be used to inject timer interrupts to the CPU.
- SW can programs bits 3-0 of the LVTTIMER register to indicate which interrupt line needs to be converted into a timer interrupt. Based on the line programmed, when the timer expires, the MVIC will inject the corresponding vector (0x20 to 0x2f). This interrupt line must be configured for edge mode.
- Bit 16 is the mask bit
- Bit 17 is the periodic mode bit.

31 18	17	16	15 8	7	6	5	4	3 0
	P	M	Reserved	0	0	1	0	Line Number

0

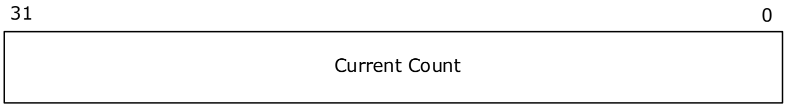
11.4.1.8 ICR

- The initial count of the timer. The timer counts down from this value to 0.
- In periodic mode, the timer automatically reloads the Current Count Register (CCR) from the ICR when the count reaches 0. At this time, the MVIC generates a timer interrupt to the core and the countdown repeats.
- If during the countdown process software writes to the ICR, counting restarts using the new initial count value.
- A write of 0 to the ICR effectively stops the local MVIC timer, in both one-shot and periodic mode.
  - The LVT Timer Register determines the vector number delivered to the core when the timer count reaches zero.
  - Software can use the mask flag in the LVT timer register to block the timer interrupt.

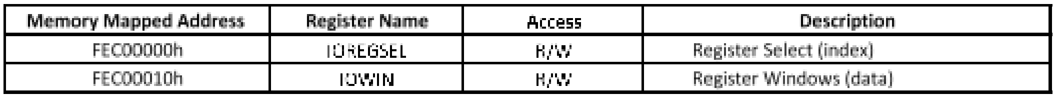


11.4.1.9 CCR

- The current count of the timer.



- Interrupt lines can be masked/unmasked and the sensitivity (level/edge) can be set by programming the registers found in the I/O controller registers

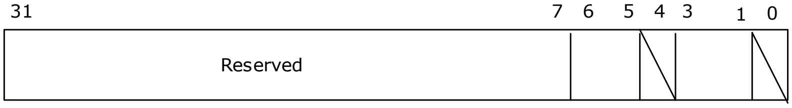


Memory Mapped Address	Register Name	Access	Description
FEC00000h	IOREGSEL	R/W	Register Select (index)
FEC00010h	IOWIN	R/W	Register Windows (data)

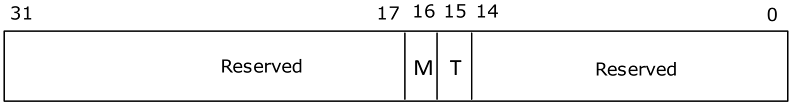
Software accesses the registers by an indirect addressing scheme using two memory mapped registers, IOREGSEL and IOWIN. Only the IOREGSEL and IOWIN registers are directly accessible in the memory address space. To setup an interrupt, software writes to IOREGSEL with a value specifying the indirect I/O register to be accessed. Software then reads or writes the IOWIN for the desired data from/to the I/O register specified by the index from the IOREGSEL. Software must access the IOWIN register as a dword quantity.

Notes:

- SW can only write to bits 1,2,3,5 and 6 of the IOREGSEL registers. Bits 0 and 4 are reserved. Example : if SW wants to access the IOWIN for index 15 {5'b01111}, then the IOREGSEL register must be programmed with value {6'b01_111_}
- SW can also read the index from the IOREGSEL register through the same bits 1,2,3,5 and 6.



- The IOWIN register has 2 bits per interrupt line (index)
  - Bit 15 : Trigger
  - Bit 16 : Mask



- Trigger: Software sets this bit to configure the interrupt signal as level sensitive. Software clears this bit to configure the interrupt signal as edge sensitive.
- Mask: Software sets this bit to mask the interrupt signal and prevent the MVIC from delivering the interrupt. The MVIC ignores interrupts signaled on a masked interrupt pin and does not deliver nor hold the interrupt pending. Changing the mask bit from unmasked to masked after the MVIC accepts the interrupt has no effect on that interrupt. When this bit is 0, the MVIC does not mask the interrupt and results in the eventual delivery of the interrupt.
- At reset, all interrupts are unmasked.

11.4.2 Programming Sequence

1. Initialize GDT
2. Initialize IDT
3. Initialize MVIC
   a. Program the respective MVIC registers.
   b. Program the mask and trigger mode for the 32 interrupt lines through the I/O registers.
4. Enables Interrupts to flow (sti)

11.4.3 Interrupt Latency Reduction

1. One INTA is issued per interrupt
   a. D2000-FST issues two INTA cycles
   b. Validation infrastructure updates required
2. The latency for the first interrupt is close to what original D2000 has. The latency to deliver subsequent interrupt of the same vector is much improved. The microcode latency is reduced to 21 cycles from 65 cycles.
   a. This optimization is only enabled in protected ring0 flat mode.
      i. CS, DS and SS base is 0 and corresponding limits are FFFF_FFFF
   b. This optimization employs an 32-entry look-aside table
      i. Entry 0 maps to external interrupt line 0 (vector 32), Entry 1 maps to external interrupt line 1 (vector 33), and so on

ii. Each entry contains a valid bit and EIP to corresponding interrupt service routine.

iii. EIP to the start of ISR is captured into the interrupt-vector lookaside buffer upon successful delivery of interrupt.

c. Invalidation of the interrupt-lookaside buffer is done when :

i. LGDT or LIDT instruction is executed

ii. Transition to non ring0

iii. When the following EFLAGS bits are set: VM, NT, RF and TF

d. ISR EIP will **not** be buffered for the following cases. Therefore, the slow interrupt delivery path will be taken if SW does the following

i. ISR uses a different CS selector than the CS selector used by the running program that is being interrupted.

ii. If the start of ISR is located outside of ITCM Instruction Flash range, i.e. 0x0018_0000 to 0x0018_7FFF.

iii. Running in real mode and not in protected mode. In real mode, the IDT is located starting at 0x0000_0000 address.

e. SW should use interrupt gate in IDT for ISR. Trap gates for ISR is not supported

f. SW should not use 16-bit segment

g. All processes running at ring-0 level and allowing interrupts should use the same pair of segments to address instructions and data to take advantage of interrupt fastpath.

3. Requirement:

a. If FW modifies IDT or GDT after LIDT or LGDT, FW must execute LIDT or LGDT again.

b. If SW wants to use “IRET/far call/far jmp” to change CS and thus descriptor, SW needs to execute LGDT or LIDT to invalidate interrupt-vector lookaside buffer first.

c. SW shall not place the start of ISR in the lowest 256 bytes, i.e. 0x0000_0000 to 0x0000_00FF.

11.4.4 Sample Code

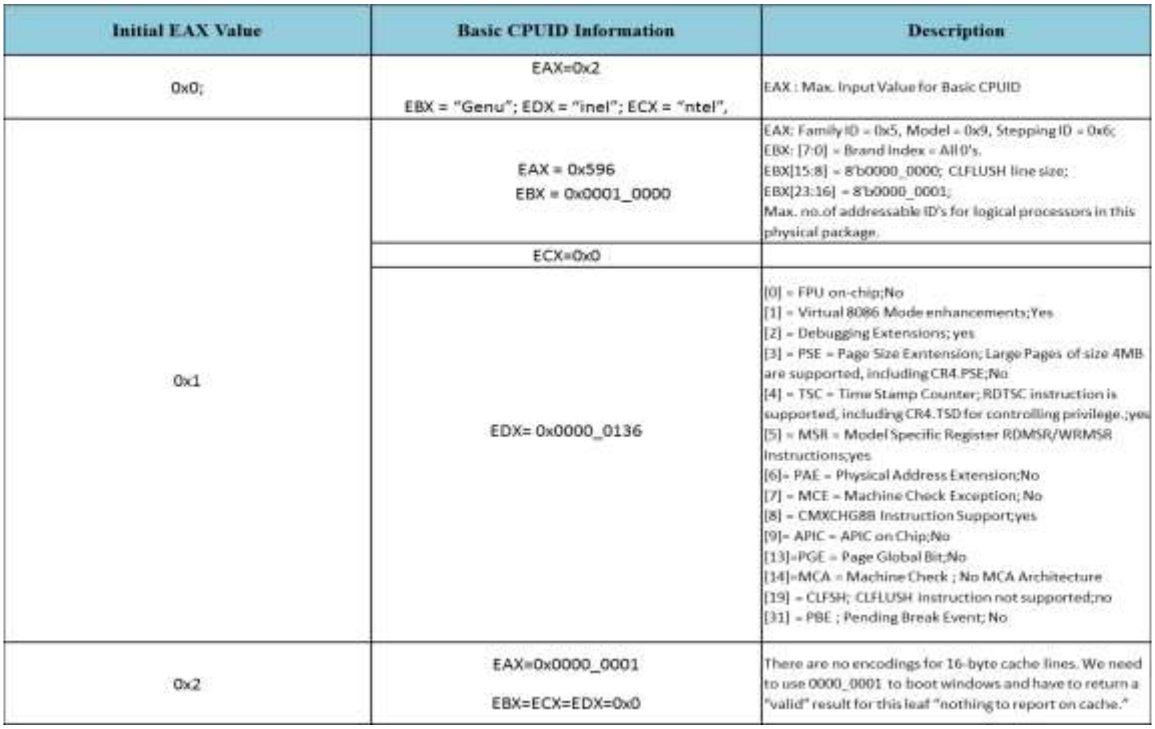
Programming timer:

```
    timer        = (unsigned int *)0xFEE00320;
    timer_cnt    = (unsigned int *)0xFEE00380;

    __asm("sti");

//  Enable Timer interrupt in Periodic mode using interrupt line 1
    *(timer)        = 0x20001;
// Timer count
    *(timer_cnt)    = 0x200;
```

Programming I/O IC registers:

```
          volatile unsigned int *rte;
    volatile unsigned int *index;

    #define LOW_NIBBLE_MASK 0x7
    #define HIGH_NIBBLE_MASK 0x18

    index        = (unsigned int *)0xFEC00000;
    rte          = (unsigned int *)0xFEC00010;

    unsigned int low_nibble;
    unsigned int high_nibble;

// Setting index in the IOREGSEL
    low_nibble   = ((line & LOW_NIBBLE_MASK)      << 0x1);
    high_nibble  = ((line & HIGH_NIBBLE_MASK)  << 0x2);

    *(index) = high_nibble | low_nibble;

// Setting Trigger mode in the IOWIN
    *(rte) = (0x1 << 15);
```

11.5 CPUID



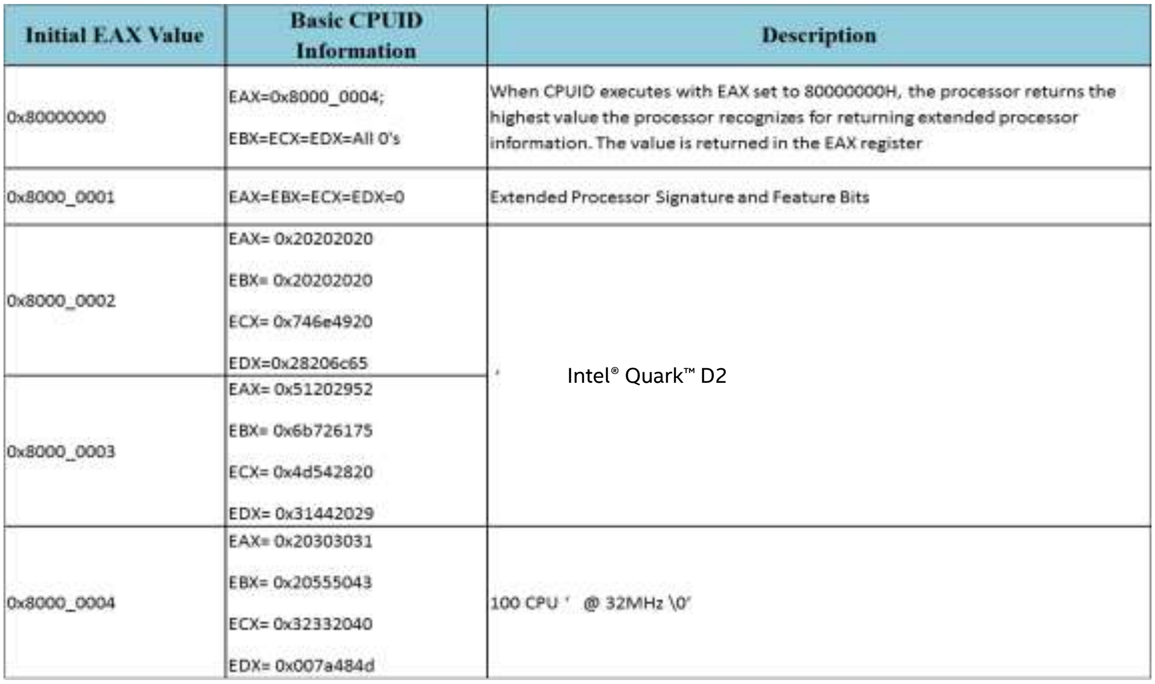
Initial EAX Value	Basic CPUID Information	Description

0x0;	EAX=0x2
	EBX = "Genu"; EDX = "inel"; ECX = "ntel",
		EAX : Max. Input Value for Basic CPUID

0x1	EAX = 0x596
	EBX = 0x0001_0000
		EAX: Family ID = 0x5, Model = 0x9, Stepping ID = 0x6;
		EBX: [7:0] = Brand Index = All 0's.
		EBX[15:8] = 8'b0000_0000; CLFLUSH line size;
		EBX[23:16] = 8'b0000_0001;
		Max. no.of addressable ID's for logical processors in this physical package.

	ECX=0x0	
	EDX= 0x0000_0136	[0] = FPU on-chip;No [1] = Virtual 8086 Mode enhancements;Yes [2] = Debugging Extensions; yes [3] = PSE = Page Size Exntension; Large Pages of size 4MB are supported, including CR4.PSE;No [4] = TSC = Time Stamp Counter; RDTSC instruction is supported, including CR4.TSD for controlling privilege.;yes [5] = MSR = Model Specific Register RDMSR/WRMSR Instructions;yes [6]= PAE = Physical Address Extension;No [7] = MCE = Machine Check Exception; No [8] = CMXCHG8B Instruction Support;yes [9]= APIC = APIC on Chip;No [13]=PGE = Page Global Bit;No [14]=MCA = Machine Check ; No MCA Architecture [19] = CLFSH; CLFLUSH instruction not supported;no [31] = PBE ; Pending Break Event; No

0x2	EAX=0x0000_0001
	EBX=ECX=EDX=0x0
		There are no encodings for 16-byte cache lines. We need to use 0000_0001 to boot windows and have to return a "valid" result for this leaf "nothing to report on cache."



Initial EAX Value	Basic CPUID Information	Description

0x80000000	EAX=0x8000_0004;
	EBX=ECX=EDX=All 0's
		When CPUID executes with EAX set to 80000000H, the processor returns the highest value the processor recognizes for returning extended processor information. The value is returned in the EAX register

0x8000_0001	EAX=EBX=ECX=EDX=0	Extended Processor Signature and Feature Bits

0x8000_0002	EAX= 0x20202020
	EBX= 0x20202020
	ECX= 0x746e4920
	EDX=0x28206c65
		Intel® Quark™ D2

0x8000_0003	EAX= 0x51202952
	EBX= 0x6b726175
	ECX= 0x4d542820
	EDX= 0x31442029	

0x8000_0004	EAX= 0x20303031
	EBX= 0x20555043
	ECX= 0x32332040
	EDX= 0x007a484d
		100 CPU ' @ 32MHz \0'

Note: Return value for EAX = 0x8000_000[2-4] does not contain "D2000" string.

12 Memory Subsystem

The memory subsystem contains the following volatile and non-volatile memories:

- System Flash – 32KB
- OTP (implemented using Flash Memory) – 8KB + 4 KB
- Internal System SRAM – 8KB

Each of these regions implement protection mechanisms with access control described later.

12.1 Features

12.1.1 System Flash Controller Features

- The Flash controller interfaces with 32KB Main Memory Block and 8KB of Information Block of Flash memory.
- Supports 64b wide reads via a dedicated Host Processor ITCM Interface.
- Supports 32b wide Instruction reads via an AHB Lite interface. The 32b reads can be performed as single read, incrementing burst or wrapping burst.
- Supports prefetching of programmable number (up to 4) 16B chunks from Flash for requests from AHB interface – prefetching can be enabled/disabled via a configuration register as part of Flash subsystem. There is no prefetching performed on ITCM interface since Processor has its own internal prefetcher.
- Supports page erase via configuration registers in the System Control Module
- Each page is 2KB.
- Supports 32b wide writes via configuration registers in the System Control Module. The procedure for updating Flash is as follows:
  - Copy Flash Page, with location to be modified, to SRAM
  - Erase the Flash page
  - Update the relevant bytes in the SRAM copy
  - Copy the modified page in SRAM into Flash
- Each 32b write operation takes approx. 40us (refer to Flash datasheet – Tnvs(5us) + Tpgs(10us) + Tprog(20us) + Tnvh(5us)). Supports optional interrupt generation capability after write completion.
- Supports mass erase via configuration registers in the System Control Module.
  - Only when 4KB Data Flash has not been programmed. It is the responsibility of FW to prevent Mass Erase when 4KB Data Flash has been programmed. HW has no built-in protection.
  - Supports optional interrupt generation capability after erase completion.
- Supports erase reference cell via configuration in the System Control Module – this needs to be done to allow reads to work – only needs to be done once in the lifetime of the device.
- Support configurable wait states to allow Flash to run with different frequency clocks. Number of wait states on Flash Read Access as follows:

- Clk = 4MHz or slower => 0 wait states supported on first access, 1 wait state on a subsequent access if the second access is on the next clock, 0 wait states on a subsequent access if there is at least a single clock cycle gap after the first access
- Clk = 8MHz => 1 wait state on all accesses
- Clk = 16MHz => 1 wait state on all accesses
- Clk = 32MHz => 2 wait states on all accesses

Clock Frequency	ITCM Latency	AHB Latency
32MHz	2 wait-states	2 wait-states
8-16MHz	1 wait-state	2 wait-states
<= 4MHz	0/1 wait-state	1 wait-state

Flash Protection mechanisms: The Flash Read Protection features are as follows:

- There are 2 Agents:
  - D2000 (determined by the AHB Master ID)
  - DMA (determined by the AHB Master ID)
    Protection mechanisms are the same between the 2 DMA channels.
- Supports a lock-out feature where Flash writes/erases are disabled via a register write. Once Flash writes have been disabled, a warm reset is required to re-enable write access.
- 4 Configurable Flash Protection Regions (FPRs) with addressing of 1KB byte granularity. There is no size restriction for each region. Each region has lower and upper bound addresses - aligned to 1KB boundary, for address range checks. Address used for comparison is offset within 36KB of Main Memory block. With 1KB alignment, only 6b are needed for lower and upper bound address comparison. The mechanism is not applicable to 8KB OTP Code region of system address space.
- The FPRs are disabled by default – in this case all Agents have RD and WR access to the whole of the Flash
- Each FPR has a set of programmable Access Control flags described as follows:

FPR Enable	D2000 RD Access Enable	DMA RD Access Enable

- The Access Control flags can be locked – when locked they can only be re-programmed after a Warm Reset
- The Regions may overlap – in this scenario ALL of the relevant Access Enables needs to be set to allow an Agent to access that overlapping region (i.e. an AND operation)
- In the event of an Access Violation during a Read Access the data is replaced with value from a config register, and the I/F protocol is handled as for a normal Read Access (the read to Flash is still performed and the read data from Flash is overwritten)
- When an Access Violation occurs, a Violation Event trigger (Interrupt) is asserted and the following information is logged: The Agent, the Address (offset within 36KB Main Memory block) and the Transfer Type (Flash RD)
- There are 2 modes for handling a Violation Event:

  - Debug Mode – the Violation Event is used to trigger Probe Mode on Processor effectively triggering a break point.
  - Normal Mode – the Violation Event is treated as an Interrupt that can be routed to the Processor.

- Support a scan mode where all the Flash control signals are gated during scan.

The FPR registers reside in the Flash Controller Configuration Registers.

The previously described protection is applied for ITCM requests also.

12.1.2 OTP Features

- 8KB OTP:
  - Implemented using the information memory region of Flash.
  - Part of Processor's Instruction address space (ITCM).
  - Supports 64b wide reads via ITCM Interface.
  - Supports 32 bit wide reads via an AHB Lite Interface.
  - A Rotated Priority access scheme is used to arbitrate between the ITCM Interface and the AHB Lite Interface.
  - Write, page erase and erase reference cell supported as per section 12.1.1 before the region is programmed as OTP.
  - Wait state behavior as per section 12.1.1 and same as Code Flash.
  - Supports a hardware lock mechanism to prevent writes to and erases of OTP. If bit 0 of offset 0x0 of the information memory region of Flash is 0b, then OTP region is considered programmed and hardware blocks all writes/erases of information memory.
  - An upper and lower lock bit to disable reads to the upper 4KB and lower 4KB regions of Flash. The register lock bits can only be cleared after a warm reset.
  - When an Access Violation occurs a Violation Event trigger (Interrupt) is asserted and the following information is logged: The Agent, the Address and the Transfer Type (ROM RD).
  - Mass Erase capability is disabled by HW once 8KB OTP has been programmed.
  - Certain offsets of information memory hold specific information as described in the table below:

32b Aligned System Address	Starting Bit Location	Size	Description
0x0000_0000	[0]	1b	8KB OTP is locked out or not. 0 --> locked out; 1 --> Writeable
0x0000_0000	[16]	16b	"Format" ID. 0x0001 is default to start with
0x0000_0004	[0]	{0,1,1,1,1,1}10b	10b trim code for 32MHz frequency for 32MHz Oscillator. This field is unique per part since it is PVT dependent
0x0000_0004	[16]	10b	10b trim code for 16MHz frequency for 32MHz Oscillator. This field is unique per part since it is PVT dependent. This field is un-programmed in MVL and defaults to all 1's.
0x0000_0008	[0]	10b	10b trim code for 8MHz frequency for 32MHz Oscillator. This field is unique per part since it is PVT dependent. This field is un-programmed in MVL and defaults to all 1's.
0x0000_0008	[16]	10b	10b trim code for 4MHz frequency for 32MHz Oscillator. This field is unique per part since it is PVT dependent. This field is un-programmed in MVL and defaults to all 1's.
0x0000_0028*	[0]	256b	Unique per part data related to Good Die Record, Wafer and die-coordinates.
0x0000_0150	[0]	Any	Reset Vector for CPU

**Note on the data contained at offset 0x00000028 – The piece of information stored in this register at time of manufacture is unique to each piece of silicon. As such, it may be considered a Unique Indentifier for a given silicon part. This information brings with it end user privacy implications – the solution designer / system integrator developing products / solutions with this part are responsible for compliance to all applicable security & privacy legislation & where they are distributing this product or solution to end customers that they are responsible for notifying those customers that they are in turn responsible for compliance to all applicable security & privacy legislation. This Unique Identifer may be removed, by reprogramming this register prior to 'locking' the OTP Flash area.*

- 4KB Data Flash:
  - There is a limitation of max 8KB of information memory inside a flash device. Hence, 4KB Data Flash is implemented using the Main Memory block of Flash. Remaining 32KB of Main Memory block is used for instruction code.
- Not part of either of Processor's ITCM or DTCM address spaces and accessed by Processor over AHB-Lite interface.
- Transparent to HW and entirely managed by FW using FPR.

- Access control is enforced via FPR.
- Supports 32 bit wide reads via an AHB Lite Interface.
- A Rotated Priority access scheme is used to arbitrate between the ITCM Interface and the AHB Lite Interface.
- FW must not issue Mass Erase once 4KB Data Flash has been programmed.

12.1.3 Internal SRAM Features

- The internal SRAM controller presents 8KB of SRAM – organized in the form of 2 banks of 4KB each.
- Supports 64 bit wide reads and writes via a dedicated Host Processor DTCM Interface. Processor only reads and writes 32b at a time. Byte enables indicate which SRAM bank the request is targeting. All accesses are 64b address aligned.
- Supports 32 bit wide reads and writes via an AHB Lite Interface.
- A Rotated Priority access scheme is used to arbitrate between the DTCM Interface and the AHB Lite Interface. The arbitration scheme will result in 0 wait states being applied to the respective interface for arbitration win scenario and 1 wait state being applied to the respective interface for arbitration loss scenario.
- Latency:

Clock Frequency	DTCM Latency	AHB Latency
All frequencies	0 wait-state	1 wait-state

Memory Region Protection capabilities are provided by the SRAM controller and apply to accesses originating from both the TCM I/F block and the AHB I/F block. These capabilities are as follows:

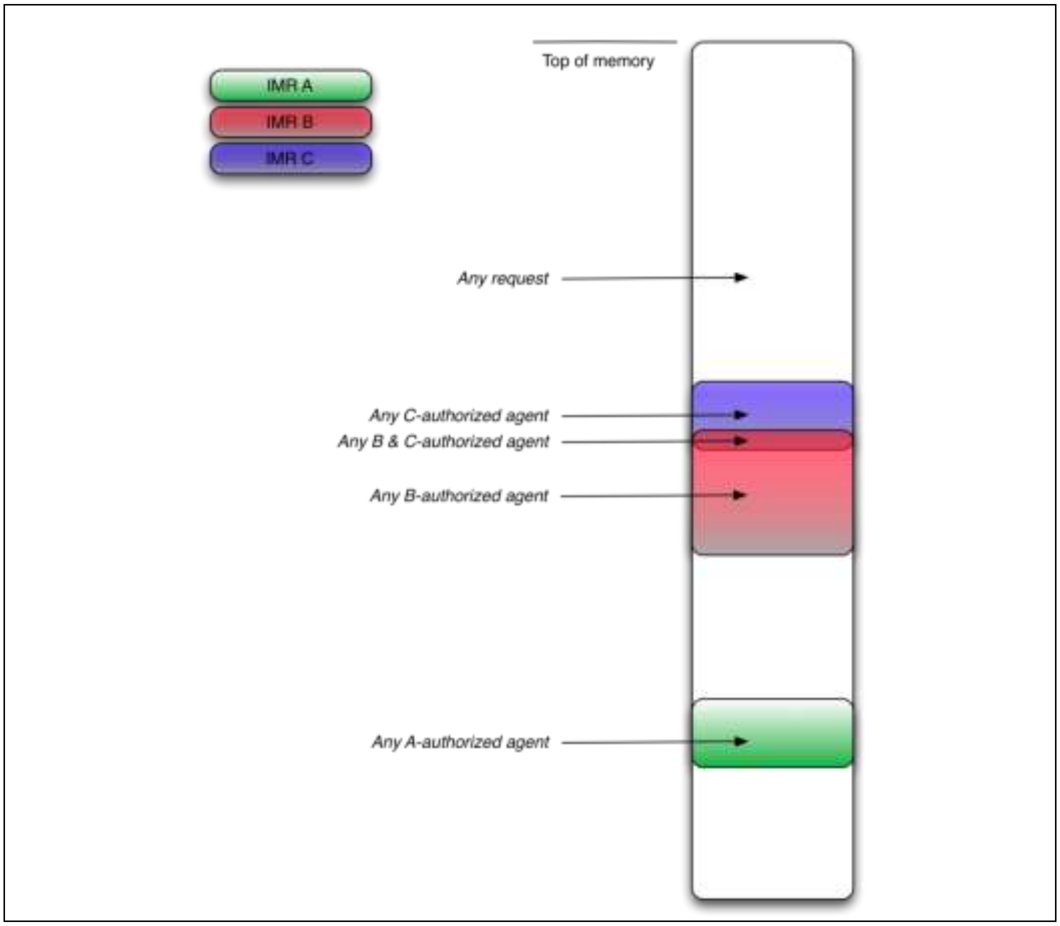
- 2 Agents:
  - D2000 Data TCM + D2000 AHB (determined by the AHB Master ID)
  - DMA
- 4 Configurable Isolated Memory Regions (IMRs) for memory protection with addressing of 1KB byte granularity
- The IMRs are defined by a lower bound and an upper bound, for an address to be within the IMR it must be greater than or equal to the lower bound **and** less than or equal to the upper bound
- The IMRs are disabled by default – in this case all Agents have RD & WR access to the whole of the SRAM
- Each IMR has a set of programmable Access Control flags described as follows:

MPR Enable	D2000 RD Access Enable	DMA RD Access Enable	D2000 WR Access Enable	DMA WR Access Enable

- The Access Control flags can be locked – when locked they can only be re-programmed after a Warm Reset
- The Regions may overlap refer to Figure 15. Example IMR zones – Requests that fall in the region overlapped by IMR B and IMR C are **only** allowed if the requesting agent is enabled for **both IMRs**
- In the event of an Access Violation during a Write Access the data is dropped, not written to SRAM and the I/F protocol is handled as for a normal Write Access (the byte enables can be brought low to satisfy this)
- In the event of an Access Violation during a Read Access the data is replaced with Dummy Data, and the I/F protocol is handled as for a normal Read Access (the read to SRAM can still be done to satisfy this with the dummy data over-riding the actual data read from SRAM)
  - This Dummy Data is programmable and is locked along with the Access Control flags above.

- When an Access Violation occurs a Violation Event trigger (Interrupt) is asserted and the following information is logged: The Agent, the Address and the Transfer Type (RD/WR)
- The Violation Event trigger is output as an interrupt - See Interrupt Routing for additional details relating to how this interrupt can be routed.

Figure 15. Example IMR zones



The IMR registers will reside in the SRAM Controller Configuration Registers.

In addition, DTCM interface also performs address range checks.

12.2 Error Handling

Scenario	Notification Event	Event Logging	Data Handling
ITCM Out of Range Read	None	None	Read data returned is 32b value from register duplicated in lower and upper DW.
ITCM Read Access Violation	Interrupt	Agent, QW Flash Address, Type 6: 8KB OTP Read (ITCM); Type 7: 36KB Flash Read (ITCM). In case of concurrent violations from ITCM and AHB, AHB is higher priority than ITCM.	Read data returned is 32b value from register duplicated in lower and upper DW.
AHB Out of Range Read/Write towards Flash	None.	None	None. AHB fabric will not decode to Flash.
AHB Read Access Violation to Flash	Interrupt	Agent, DW Flash Address, Type. In case of concurrent violations from ITCM and AHB, AHB is higher priority than ITCM.	Read data returned is 32b value from register.
AHB Write to Flash Address Space	Interrupt	Agent, DW Flash Address, Type. In case of concurrent violations from ITCM and AHB, AHB is higher priority than ITCM.	Write Data is dropped
AHB Write Access Violation to Flash i.e. Program access when Write is disabled	Interrupt	Agent, DW Flash Address, Type. In case of concurrent violations from ITCM and AHB, AHB is higher priority than ITCM.	Write transaction is not initiated.
AHB Program Address is Out of Range to 8KB OTP and 36KB Flash	None	None	Write transaction is not initiated.
DTCM Out of Range Read	None	None	Read data returned is 32b value from register duplicated in lower and upper DW.

Scenario	Notification Event	Event Logging	Data Handling
DTCM Read Access Violation	Interrupt	Agent, QW Address (bit2 undefined), Type. Violations from DTCM and AHB are mutually exclusive.	Read data returned is 32b value from register.
DTCM Out of Range Write	None	None	Write data sent to SRAM with inactive BEs.
DTCM Write Access Violation	Interrupt	Agent, QW Address (bit2 undefined), Type. Violations from DTCM and AHB are mutually exclusive.	Read data returned is 32b value from register duplicated in lower and upper DW. Write data sent to SRAM with inactive BEs.
AHB Out of Range Read/Write towards SRAM	None.	None	None. AHB fabric will not decode to SRAM.
AHB Read Access Violation to SRAM	Interrupt	Agent, DW Address (bit2 undefined), Type. Violations from DTCM and AHB are mutually exclusive.	Read data returned is 32b value from register.
AHB Write Access Violation to SRAM	Interrupt	Agent, DW Address (bit2 undefined), Type. Violations from DTCM and AHB are mutually exclusive.	Write data sent to SRAM with inactive BEs.

Behavior of CPU prefetch crossing address boundaries is as follows:

- Case 1: Reset vector fetch from 0xFFFF_FFF0 and 0xFFFF_FFF8 can potentially cause the CPU to prefetch from address 0x0, 0x8 and 0x10
  - The accesses to 0x0, 0x8, and 0x10 fall in the OTP code region. SoC will return the data from the OTP ROM.
  - This will NOT be treated as an error condition and No spurious interrupts will be sent to the CPU.
- Case 2: CPU executing from 0x1FFF (end of OTP Instruction ROM). CPU prefetch can spill over to the reserved range 0x2000-0x7_FFFF.
  - The accesses to the reserved range will be completed. The SoC will return a fixed value of all Cs.
  - This will NOT be treated as an error condition and No spurious interrupts will be sent to the CPU.
- Case 3: CPU executing from 0x18_7FFF (end of Instruction RAM). CPU prefetch can spill over to the reserved range 0x18_8000-0x1F_FFFF.
  - The accesses to the reserved range will be completed. The SoC will return a fixed value of all Cs.

  - This will NOT be treated as an error condition and No spurious interrupts will be sent to the CPU.

- Case 4: CPU executing from 0x20_0FFF (end of Data ROM). CPU prefetch can spill over to the reserved range 0x20_1000-0x27_FFFF.
  - The accesses to the reserved range will be completed.
  - Since Data ROM access are sent out on AHB. The AHB fabric will return an error and can inject an interrupt for error notification.

- Case 5: CPU executing from 0x28_1FFF (end of Data RAM). CPU prefetch can spill over to the reserved range 0x28_2000-0x2F_FFFF.
  - The accesses to the reserved range will be completed. The SoC will return a fixed value of all Cs.
  - Instruction fetches to the Data RAM are sent out on AHB Interface. The AHB fabric will return an error and can inject an interrupt for error notification.

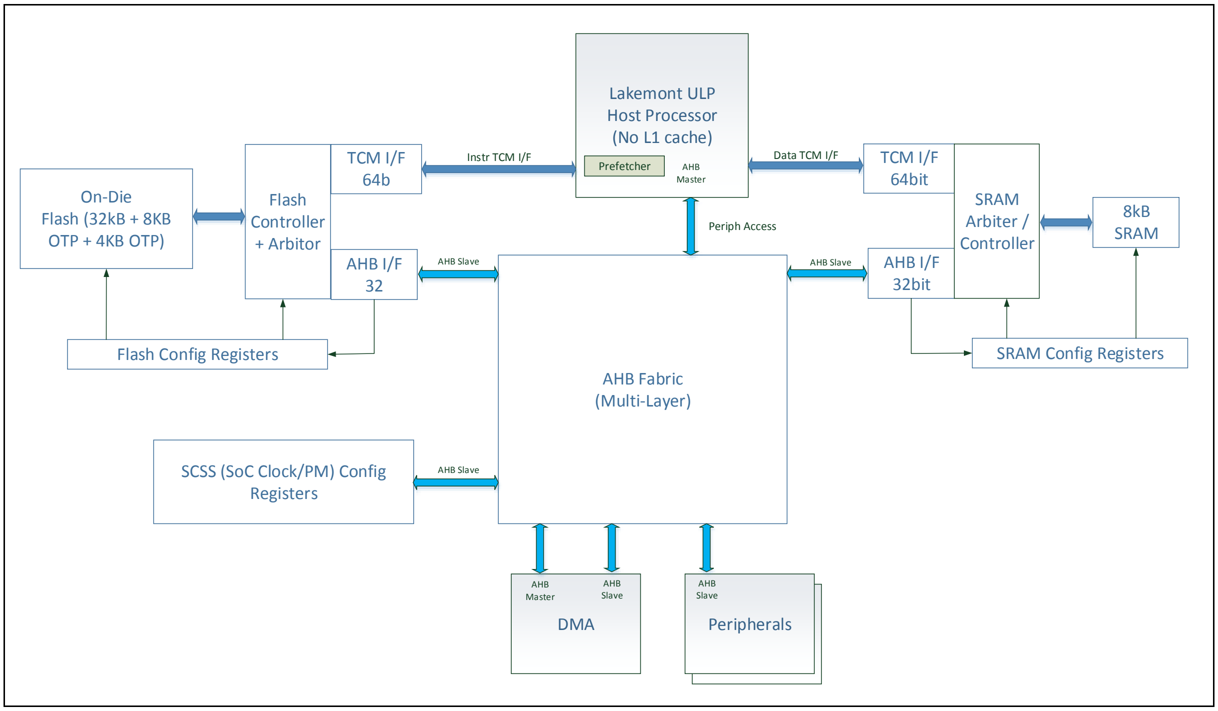
- Case 6: CPU executing from a specific region of Data RAM which is setup via IMR (Isolated Memory Region) registers such that CPU has privileges to execute from that region. CPU prefetch can spill over to adjacent region for which CPU does not have privileges to execute.
  - The accesses to violated region will complete.
  - Instruction fetches to the Data RAM are sent out on AHB Interface. SRAM controller will flag an Access Control violation and can inject an interrupt for error notification.

For cases 4, 5 and 6, the recommendation is for firmware to place the code at the start of the Data RAM to avoid these scenarios and the potential spurious interrupts.

12.3 Memory Consistency Analysis

The following illustration is the block diagram of the memory subsystem.

Figure 16. Block Diagram of The Memory Subsystem



From memory consistency analysis perspective, Intel® Quark™ microcontroller D2000 consists of following key IPs:

5. ULP Quark CPU core
6. Flash Memory & Flash Controlller
7. SRAM & SRAM Controlller
8. AHB/APB Fabric
9. DMA
10. Slave Peripherals
11. SCSS Configuration Registers

- D2000-ULP:
    - Pentium x86 ISA
    - No I-Cache or D-Cache and hence no need for snoops.
    - 32b Addressing
    - C0, C1 and C2 power states
    - No FPU
    - No support for Atomic operations
    - 3 key interfaces:
        - 64b ITCM: Instruction TCM

- Address range hardwired – see D2000-ULP Address Map later
- Code Reads to ITCM address range
- Aggressive prefetcher for instruction fetches
- Wait-state capability
- Fastest access limited by eFlash latency: 2 wait-state @ 32MHz

- 64b DTCM: Data TCM
  - Address range hardwired – see D2000-ULP Address Map later
  - Reads/Writes to DTCM address range
  - Wait-state capability
- 32b AHB-Lite: Peripheral interface
  - Master on AHB Lite interface
  - No AHB Slave port
  - No bursting capability
  - Code Reads to DTCM address range
  - Data reads/writes to ITCM address range
  - All probe mode accesses are issued on AHB
  - I/O reads/writes are unsupported by the Intel® Quark™ microcontroller D2000. Such transactions are under SW control and aliased to Memory range on AHB-Lite.
  - All transactions are non-posted transactions.
  - Atomic/Locked transactions are issues as regular memory transactions without lock semantics.

- IO(x)APIC + LAPIC
  - 32 IRQs
- Memory Ordering Model
  - Processor Ordered Memory model
  - DTCM reads are allowed to go ahead of writes (to different addresses than reads)
  - All AHB transactions are in program order.
  - Transactions across ITCM and DTCM interfaces have no ordering relationship to each other.
  - Transactions across ITCM and AHB-Lite interfaces have no ordering relationship to each other. In other words, ITCM read transactions can occur concurrently with AHB-Lite reads/writes.
  - All Stores in D2000 must be in program order. It implies:
    - Writes on DTCM interface must be in program order.
    - Writes on AHB Lite interface must be in program order.
    - DTCM write (data access) followed by AHB write (data access) must be in program order.
      - Specifically, the AHB write cannot start until DTCM write completes. An implication of this is that if SoC introduces wait-states on DTCM interface for the write, D2000 will not start the AHB write until the DTCM write completes i.e. wait state goes away and SRAM controller accepts the write.
    - AHB write (data access) followed by DTCM write (data access) must be in program order.
      - Similar to previous case. The DTCM write cannot start until AHB write completes. An implication of this is that if SoC introduces wait-states on AHB-Lite interface for the write, D2000 will not start the

DTCM write until the AHB write completes i.e. AHB fabric gives a response for the write.

- All DTCM writes and UC AHB reads (data accesses) must be in program order.
    - Same as A3 and A4 above except replace AHB write with AHB read.
    - AHB reads that are code accesses (e.g. to SRAM) have no ordering relationship with DTCM writes.
- Inside D2000, there is only a single data access outstanding at any time

- Flash Memory & Flash Controller:
    - Flash controller has 2 interfaces: ITCM and AHB Slave.
    - Flash Memory has asynchronous interface i.e. there is no clock input.
    - Read from embedded Flash memory is similar to a standard SRAM read but with increased latency.
    - However, there is no write capability to Flash memory that is equivalent to a standard SRAM write.
    - Flash allows a 32b write to arbitrary location (Program Operation) that takes ~40us (36KB Flash Memory size). There is an optional interrupt generation capability after write completion. However, only a '1' in a cell can be changed to a '0' during a Program operation. The only mechanism to write a '1' to a location is to perform a Page–erase operation (2KB page size in Intel® Quark™ microcontroller D2000) or a Mass erase operation which erases entire flash contents to all 1's. An erase operation lasts for ~20ms duration. There is an optional interrupt generation capability after erase completion.
    - A Program operation is performed via a sequence of writes to Flash Configuration registers.
        - Write target 32b data content to a register
        - Write target flash address to a register
        - Write to a bit in a control register that triggers the required Program sequence operation.
    - Flash Configuration registers reside off of AHB fabric sharing the AHB Slave port with Flash Controller/Memory.
    - There is also an upper bound on how many program operations can be performed to same cell - in between erases.
    - Flash Memory has a single port and only a single operation can be performed at a time. During flash program or erase operation, flash memory is not accessible during the entire duration.
    - The AHB Slave port has no write-posting capability.
    - Flash latency can be changed to optimize latency for different clock frequencies.
- SRAM & SRAM Controller:
    - SRAM controller has 2 interfaces: DTCM and AHB Slave.
    - Standard SRAM reads and writes
    - SRAM Configuration registers reside off of AHB fabric sharing the AHB Slave port with SRAM Controller/Memory.
    - The AHB Slave port and DTCM interface have no write-posting capability.
- AHB/APB Fabric
    - No write posting support (that is, all writes are non-posted).
    - In-order fabric with no pipelining.
    - Only 2 masters on fabric: D2000 and DMA.
    - No locked transaction support.
    - One fabric interface for every master and slave.

- DMA
  - Separate Master and Slave interfaces on AHB interface
  - No write posting support (that is, all writes are non-posted).
- Slave Peripherals
  - Examples: UART, I^{2}C, SPI, and so on.
  - No write posting support (that is, all writes are non-posted).
  - Single AHB slave interface to each peripheral.
- SCSS Configuration Registers
  - Control SoC configuration related to Clocks, Power Management, and so on.
  - Reside on AHB fabric on a single AHB slave interface

12.3.1 Producer/Consumer Model Analysis of the Memory Subsystem

Flash Memory is a special case and is discussed later.

From standard Producer/Consumer model perspective, producer produces data, producer writes/triggers flag, consumer reads/receives flag, and consumer consumes data. Goal is to ensure consumer's data is consistent.

- Producers can be: DMA, D2000
- Consumers can be: DMA, D2000
- Data can reside in: SRAM, AHB (DMA/Registers on AHB fabric)
- Flag: D2000 Interrupt, SRAM, AHB (DMA/Registers on AHB fabric)

The following scenario combinations are possible and analyzed – some of them are illegal use models but analyzed for completeness.

Producer	Consumer	Data	Flag	Analysis
DMA	DMA	SRAM, AHB	SRAM, AHB or D2000 Interrupt	DMA is in-order and does no write-posting. AHB fabric is in-order. All writes in AHB fabric are non-posted. Hence, no memory consistency concerns.
DMA	D2000	SRAM	SRAM, AHB or D2000 Interrupt	DMA is in-order and does no write-posting. AHB fabric is in-order. All writes in AHB fabric are non-posted. Hence, when DMA completes, data is guaranteed to be present in SRAM. So when D2000 reads SRAM via DTCM interface, data is consistent.
		AHB	SRAM, AHB or D2000 Interrupt	DMA is in-order and does no write-posting. AHB fabric is in-order. All writes in AHB fabric are non-posted. Hence, when DMA completes, data is guaranteed to be present in AHB. So when D2000 reads flag on SRAM and then reads data via AHB interface, data is consistent.
D2000	DMA	SRAM	SRAM	DTCM writes from D2000 are in-order and in program order. Since AHB fabric is in-order, DMA reads will observe data in same order. Hence, no memory consistency concerns.

Producer	Consumer	Data	Flag	Analysis
		SRAM	AHB, D2000 Interrupt	There is no posting buffer in SRAM controller on DTCM interface. D2000 ensures DTCM write completes before AHB write (to DMA) is issued. Hence, DMA read will receive consistent data when it reads SRAM via AHB.
		AHB	SRAM	D2000 ensures AHB write completes before DTCM write is issued. Hence, DMA read will receive consistent data when it reads data on AHB.
		AHB	AHB, D2000 Interrupt	Since D2000 writes to AHB are in-order and in program order, there is no memory consistency concern.
D2000	D2000	SRAM	SRAM, D2000 Interrupt	D2000 is self-consistent.
		SRAM	AHB	D2000 ensures DTCM write completes before AHB write is issued.
		AHB	SRAM	D2000 ensures AHB write completes before DTCM write is issued.
		AHB	AHB, D2000 Interrupt	Since D2000 writes to AHB are in-order and in program order, there is no memory consistency concern.

Write to flash memory write has a very restricted use model. Flash write can be performed for following use cases since SRAM is used for updating run-time data by an application:

- Initialize code (e.g. boot code) after manufacturing,
- Update FW
- Load App

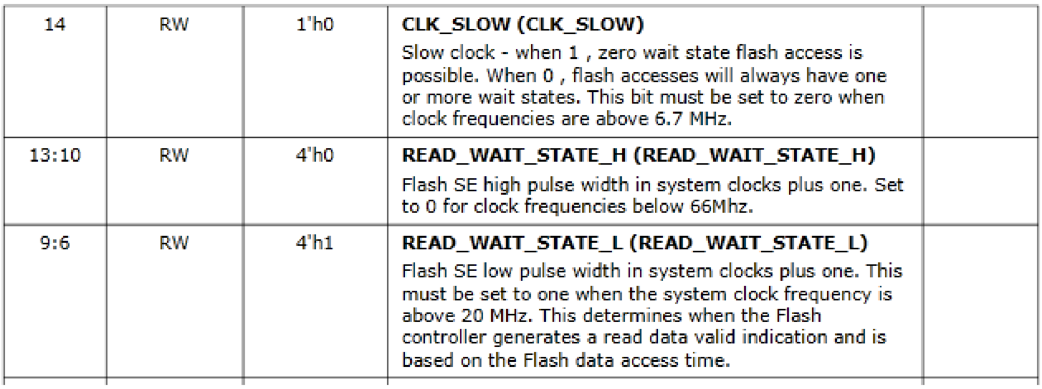
Only D2000 can initiate a write to Flash memory via a sequence of register writes to Flash Configuration registers. The only two interesting cases are:

- Handling of SW breakpoint: SW tool chain must perform following sequence of operations:
  - Copy the appropriate 2K page (that contains the set of instructions to be modified) from Flash memory to SRAM.
  - Copy the original set of instructions in SRAM to a temporary location (for later restoration) in SRAM
  - Update 2K page in SRAM with new instructions
  - Erase the 2KB page in Flash
  - In a loop, program flash memory 32b at a time.
  - When the loop completes, flash memory has updated 2K page.
  - Since prefetcher depth is only 128b and a long sequence of operations need to be performed to program flash memory, D2000's prefetcher cannot contain stale instructions.
- Interaction of D2000's instruction prefetcher vs Flash memory updates: From Intel® Quark™ microcontroller D2000's usage model perspective, the routines to program/erase flash memory are part of boot code. It is an illegal usage model to modify this boot code while simultaneously executing from it. Hence, this is not a concern.

12.3.2 Miscellaneous Memory Ordering related Scenarios

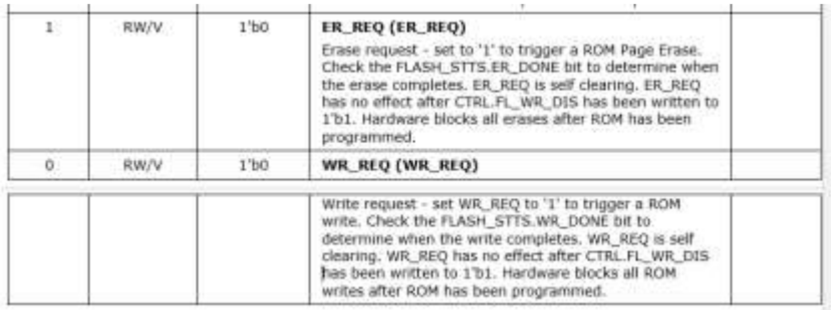
Intel® Quark™ microcontroller D2000 has registers on AHB fabric that can alter configurations of various functions of the SoC. Some examples are:

- Flash Latency: These fields can be changed by SW to alter flash latency dynamically based on SoC clock frequency.



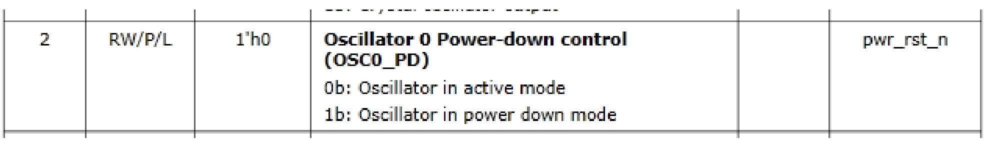
14	RW	1'h0	**CLK_SLOW (CLK_SLOW)** Slow clock - when 1 , zero wait state flash access is possible. When 0 , flash accesses will always have one or more wait states. This bit must be set to zero when clock frequencies are above 6.7 MHz.	
13:10	RW	4'h0	**READ_WAIT_STATE_H (READ_WAIT_STATE_H)** Flash SE high pulse width in system clocks plus one. Set to 0 for clock frequencies below 66Mhz.	
9:6	RW	4'h1	**READ_WAIT_STATE_L (READ_WAIT_STATE_L)** Flash SE low pulse width in system clocks plus one. This must be set to one when the system clock frequency is above 20 MHz. This determines when the Flash controller generates a read data valid indication and is based on the Flash data access time.	

- Flash Program/Erase operation: Program/Erase is under SW control.



1	RW/V	1'b0	**ER_REQ (ER_REQ)** Erase request - set to '1' to trigger a ROM Page Erase. Check the FLASH_STTS.ER_DONE bit to determine when the erase completes. ER_REQ is self clearing. ER_REQ has no effect after CTRL.FL_WR_DIS has been written to 1'b1. Hardware blocks all erases after ROM has been programmed.	
0	RW/V	1'b0	**WR_REQ (WR_REQ)**	
			Write request - set WR_REQ to '1' to trigger a ROM write. Check the FLASH_STTS.WR_DONE bit to determine when the write completes. WR_REQ is self clearing. WR_REQ has no effect after CTRL.FL_WR_DIS has been written to 1'b1. Hardware blocks all ROM writes after ROM has been programmed.	

- System Clock/Oscillator Control: SW can power down the main oscillator and go to deep-sleep state only to be woken up by an external wake event via analog comparator input. The wake event requires no SoC clock and it enables the system clock/oscillator.



2	RW/P/L	1'h0	**Oscillator 0 Power-down control (OSC0_PD)** 0b: Oscillator in active mode 1b: Oscillator in power down mode		pwr_rst_n

Since AHB write to above register fields can be concurrent to ITCM code fetches, some form of synchronization is required to ensure ITCM interface is quiesced for these registers to alter SoC operation in a safe manner. It is possible to analyze the above 3 cases and find solutions. But a capability to control ITCM prefetching under specific conditions is a very useful capability that is lacking. One of the solutions is to execute HALT instruction as a following instruction that updates above register fields. Upon completion of HALT instruction, D2000 is guaranteed to have made all interfaces idle. The assertion of HALT signal (output of D2000) can be used as a deterministic mechanism to take effect the new states of registers like above.

12.4 Memory Mapped IO Registers

This section describes IO registers.

12.4.1 Flash Controller 0 Register Summary

MEM BaseAddress: 0xB0100000

MEM Address	Default	Name
0x0	0000_0060h	TMG_CTRL
0x4	0000_0000h	ROM_WR_CTRL
0x8	0000_0000h	ROM_WR_DATA
0xC	0000_0000h	FLASH_WR_CTRL
0x10	0000_0000h	FLASH_WR_DATA
0x14	0000_0000h	FLASH_STTS
0x18	0000_0000h	CTRL
0x1C	0000_0000h	FPR0_RD_CFG
0x20	0000_0000h	FPR1_RD_CFG
0x24	0000_0000h	FPR2_RD_CFG
0x28	0000_0000h	FPR3_RD_CFG
0x2C	0000_0000h	MPR_WR_CFG
0x30	0000_0000h	MPR_VSTS
0x34	CCCC_CCCCh	MPR_VDATA

12.4.2 Flash Controller 0 Register Detailed Description

12.4.2.1 TMG_CTRL (TMG_CTRL)

Flash Timing Control Register. There is a SW programming restriction for this register. When switching SoC to a higher frequency, this register must be updated first to reflect settings associated with higher frequency **BEFORE** SoC frequency is changed. On the other hand, when switching SoC to a lower frequency, this register must be updated only 6 NOP instructions **AFTER** the SoC frequency has been updated. Otherwise, flash timings will be violated.

READ_WAIT_STATE_L and MICRO_SEC_CNT are optimized for 32MHz. These settings are conservative for lower frequency and add access latency for reads. But functionally, these settings will work for any frequency <= 32MHz for reads. For program/erase operations, default value of MICRO_SEC_CNT will violate flash timings at < 32MHz frequency. Hence, MICRO_SEC_CNT must be changed before flash can be programmed/erased at non-32MHz frequency.

MEM Offset (B0100000)	0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0060h

Bits	Access Type	Default	Description
31:15	RO	17'h0	**RSV** Reserved
14	RW	1'h0	**CLK_SLOW** Slow clock - when 1 , zero wait state flash access is possible. When 0 , flash accesses will always have one or more wait states. This bit must be set to zero when clock frequencies are above 6.7 MHz.
13:10	RW	4'h0	**READ_WAIT_STATE_H** Upper bit (i.e. bit 13) is **32MHz_Exit_Latency_Opt** Setting of this bit reduces the potential 128 clock latency. FW is expected to set this bit during boot. Remaining 3 bits: high pulse width in system clocks plus one. Set to 0 for clock frequencies below 66MHz.
9:6	RW	4'h1	**READ_WAIT_STATE_L** Flash SE low pulse width in system clocks plus one. This must be set to one when the system clock frequency is above 20 MHz. This determines when the Flash controller generates a read data valid indication and is based on the Flash data access time.
5:0	RW	6'h20	**MICRO_SEC_CNT** Number of clocks in a micro second.

12.4.2.2 ROM_WR_CTRL (ROM_WR_CTRL)

ROM Write Control Register

This register is only applicable for 8KB OTP region. There is no equivalent register for 4KB Data Flash region.

Before issuing flash erase/program operation,

1. FW must disable all interrupts except the flash interrupt that indicates completion of erase/program operation.
2. Issue the MMIO write that triggers the program or erase operation.
3. Issue HALT instruction.

As part of program/erase completion ISR, interrupts can be re-enabled.

MEM Offset (B0100000)	4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:20	RO	12'h0	**RSV** Reserved
19:2	RW	18'b0	**WR_ADDR** Write Address Upper 2 address bits are unused by HW. SW must ensure there is no address aliasing. Address must be in Flash Physical Address Space.
1	RW/V	1'b0	**ER_REQ** Erase request - set to '1' to trigger a ROM Page Erase. Check the FLASH_STTS.ER_DONE bit to determine when the erase completes. ER_REQ is self clearing. ER_REQ has no effect after CTRL.FL_WR_DIS has been written to 1'b1. Hardware blocks all erases after ROM has been programmed.
0	RW/V	1'b0	**WR_REQ** Write request - set WR_REQ to '1' to trigger a ROM write. Check the FLASH_STTS.WR_DONE bit to determine when the write completes. WR_REQ is self clearing. WR_REQ has no effect after CTRL.FL_WR_DIS has been written to 1'b1. Hardware blocks all ROM writes after ROM has been programmed.

12.4.2.3 ROM_WR_DATA (ROM_WR_DATA)

ROM Write Data
This register is only applicable for 8KB OTP region. There is no equivalent register for 4KB Data Flash region.

MEM Offset (B0100000)	8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**DATA** ROM Write Data

12.4.2.4 FLASH_WR_CTRL (FLASH_WR_CTRL)

Flash Write Control Register

Before issuing flash erase/program operation:

1. FW must disable all interrupts except the flash interrupt that indicates completion of erase/program operation.
2. Issue the MMIO write that triggers the program or erase operation.
3. Issue HALT instruction.

As part of program/erase completion ISR, interrupts can be re-enabled.

MEM Offset (B0100000)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:26	RO	6'h0	**RSV** Reserved
25	RW	1'b0	**INTR_EN_ER** Enable interrupt after Erase completion. This bit has no effect on ER_DONE status bit.
24	RW	1'b0	**INTR_EN_WR** Enable interrupt after Program cycle completion. This bit has no effect on WR_DONE status bit.
23:20	RO	4'b0	**RSV** Reserved
19:2	RW	18'b0	**WR_ADDR** Write Address Upper 2 address bits are unused by HW. So, SW must ensure there is no address aliasing. Address must be in Flash Physical Address Space.
1	RW/V	1'b0	**ER_REQ** Erase request - set to '1' to trigger a Flash Page Erase. The page number is specified by WR_ADDR[17:11]. ER_REQ is self clearing. ER_REQ has no effect after CTRL.FL_WR_DIS has been written to '1'.
0	RW/V	1'b0	**WR_REQ** Write request - set WR_REQ to '1' to trigger a Flash write. Check the FLASH_STTS.WR_DONE bit to determine when the write completes. WR_REQ is self clearing. WR_REQ has no effect after CTRL.FL_WR_DIS has been written to '1'.

12.4.2.5 FLASH_WR_DATA (FLASH_WR_DATA)

Flash Write Data

MEM Offset (B0100000)	10h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**DATA** Flash Write Data

12.4.2.6 FLASH_STTS (FLASH_STTS)

Flash Status

MEM Offset (B0100000)	14h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'h0	**RSV** Reserved
2	RO/V	1'h0	**ROM_PROG** 8KB ROM programmed - when set this indicates that ROM has been programmed and any further attempt to write ROM is blocked.
1	RO/C/V	1'b0	**WR_DONE** Write done - when set this indicates that a write operation has completed. WR_DONE is cleared on read.
0	RO/C/V	1'h0	**ER_DONE** Erase done - when set this indicates that an erase has completed. ER_DONE is cleared on read.

12.4.2.7 CTRL (CTRL)

Control Register

ROM below refers to 8KB OTP region only.

MEM Offset (B0100000)	18h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:9	RO	23'h0	**RSV** Reserved
18:16	RW	3'h0	**PRF_CNT** Number of 8B entries to be prefetched.
15:9	RO	7'h0	**RSV** Reserved
8	RW/V	1'h0	**ERC**

Bits	Access Type	Default	Description
7	RW/V	1'h0	**MASS_ERASE** Mass Erase - set to '1' to trigger an erase of Flash. This has no effect after FL_WR_DIS has been written to '1'.
6	RW	1'h0	**MASS_ERASE_INFO** Mass Erase Info - Valid when MASS_ERAES is '1'. When MASS_ERASE_INFO = '1' then the ROM portion of Flash is erased during a Mass Erase. When MASS_ERASE_INFO = '0' then the ROM portion of Flash is not erased during a Mass Erase. If the ROM portion of Flash is detected as programmed by the Flash Controller then hardware will block a ROM erase. This has no effect after FL_WR_DIS has been written to '1'.
5	RW	1'h0	**LVE_MODE** Low Voltage mode
4	RW/1S	1'h0	**FL_WR_DIS** Flash Write Disable
3	RW/1S	1'h0	**ROM_RD_DIS_U** Rom read disable for upper 4k region of ROM
2	RW/1S	1'h0	**ROM_RD_DIS_L** Rom read disable for lower 4k region of ROM
1	RW	1'h0	**PRE_FLUSH** Prefetch buffer Flush
0	RW	1'h0	**PRE_EN** Prefetch Enable. When '1', prefetching is enabled. When '0', prefetching is disabled.

12.4.2.8 FPR0_RD_CFG (FPR0_RD_CFG)

Flash Protection Region Read Control Register 0

MEM Offset (B0100000)	1Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'h0	**LOCK** Lock the Memory Protection Region Configuration.
30	RW/L	1'h0	**ENABLE** Enable the Memory Protection Region.
29:24	RO	6'h0	**RSV2** Reserved

Bits	Access Type	Default	Description
23:20	RW/L	4'h0	**RD_ALLOW** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA [3] : Reserved
19:18	RO	2'h0	**RSV1** Reserved
17:10	RW/L	8'h0	**UPR_BOUND** Upper 2 address bits of this register are unused by HW. The Upper Address Bound is compared with incoming Flash Address [15:10] to determine the upper 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 1's. Hence, incoming address is checked to be **less than or equal to** this field.
9:8	RO	2'h0	**RSV** Reserved
7:0	RW/L	8'h0	**LWR_BOUND** Upper 2 address bits of this register are unused by HW. The Lower Address Bound is compared with incoming Flash Address [15:10] to determine the lower 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 0's. Hence, incoming address is checked to be **greater than or equal to** this field.

12.4.2.9 FPR1_RD_CFG (FPR1_RD_CFG)

Flash Protection Region Read Control Register 1

MEM Offset (B0100000)	20h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'h0	**LOCK** Lock the Memory Protection Region Configuration.
30	RW/L	1'h0	**ENABLE** Enable the Memory Protection Region.

Bits	Access Type	Default	Description
29:24	RO	6'h0	**RSV2** Reserved
23:20	RW/L	4'h0	**RD_ALLOW** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA [3] : Reserved
19:18	RO	2'h0	**RSV1** Reserved
17:10	RW/L	8'h0	**UPR_BOUND** Upper 2 address bits of this register are unused by HW. The Upper Address Bound is compared with incoming Flash Address [15:10] to determine the upper 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 1's. Hence, incoming address is checked to be **less than or equal to** this field.
9:8	RO	2'h0	**RSV** Reserved
7:0	RW/L	8'h0	**LWR_BOUND** Upper 2 address bits of this register are unused by HW. The Lower Address Bound is compared with incoming Flash Address [15:10] to determine the lower 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 0's. Hence, incoming address is checked to **be greater than or equal to** this field.

12.4.2.10 FPR2_RD_CFG (FPR2_RD_CFG)

Flash Protection Region Read Control Register 2

MEM Offset (B0100000)	24h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'h0	**LOCK** Lock the Memory Protection Region Configuration.
30	RW/L	1'h0	**ENABLE** Enable the Memory Protection Region.

Bits	Access Type	Default	Description
29:24	RO	6'h0	**RSV2** Reserved
23:20	RW/L	4'h0	**RD_ALLOW** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA [3] : Reserved
19:18	RO	2'h0	**RSV1** Reserved
17:10	RW/L	8'h0	**UPR_BOUND** Upper 2 address bits of this register are unused by HW. The Upper Address Bound is compared with incoming Flash Address [15:10] to determine the upper 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 1's. Hence, incoming address is checked to be **less than or equal to** this field.
9:8	RO	2'h0	**RSV** Reserved
7:0	RW/L	8'h0	**LWR_BOUND** Upper 2 address bits of this register are unused by HW. The Lower Address Bound is compared with incoming Flash Address [15:10] to determine the lower 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 0's. Hence, incoming address is checked to be **greater than or equal to** this field.

12.4.2.11 FPR3_RD_CFG (FPR3_RD_CFG)

Flash Protection Region Read Control Register 3

MEM Offset (B0100000)	28h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'h0	**LOCK** Lock the Memory Protection Region Configuration.
30	RW/L	1'h0	**ENABLE** Enable the Memory Protection Region.

Bits	Access Type	Default	Description
29:24	RO	6'h0	**RSV2** Reserved
23:20	RW/L	4'h0	**RD_ALLOW** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA [3] : Reserved
19:18	RO	2'h0	**RSV1** Reserved
17:10	RW/L	8'h0	**UPR_BOUND** Upper 2 address bits of this register are unused by HW. The Upper Address Bound is compared with incoming Flash Address [15:10] to determine the upper 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 1's. Hence, incoming address is checked to be **less than or equal to** this field.
9:8	RO	2'h0	**RSV** Reserved
7:0	RW/L	8'h0	**LWR_BOUND** Upper 2 address bits of this register are unused by HW. The Lower Address Bound is compared with incoming Flash Address [15:10] to determine the lower 1KB aligned value of the Protected Region. For address comparison purposes, lower bits[9:0] are assumed to be all 0's. Hence, incoming address is checked to be **greater than or equal to** this field.

12.4.2.12 MPR_WR_CFG (MPR_WR_CFG)

Flash Write Protection Control Register

This register is unused in HW and serves no purpose.

MEM Offset (B0100000)	2Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'h0	**Valid** Prevent further writes to Flash if the register is used.

Bits	Access Type	Default	Description
30:0	RO	31'h0	**RSV** Reserved

12.4.2.13 MPR_VSTS (MPR_VSTS)

Protection Status Register

MEM Offset (B0100000)	30h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1C/V	1'h0	**VALID** This field is asserted when a Violation Event Occurs. No further Violation Events will be captured in this register until this bit is cleared by SW.
30:23	RO	8'h0	**RSV** Reserved
22:21	RO/V	2'h0	**AGENT** This field captures the Agent ID for an Access Violation - this field is valid for AHB violations only. 0: Host Processor 1: Reserved 2: DMA Engine 3: Reserved
20:18	RO/V	3'h0	**TYPE** This field captures the Transfer Type for an Access Violation: 0: ROM Read (via AHB) 1: ROM Write (via AHB) 2: Flash Read (via AHB) 3: Flash Write (via AHB) 4: Flash Write/erase (via CREG) 5: ROM Write/erase (via CREG) 6: 8KB OTP Read (via ITCM) 7: 32KB Flash Read (via ITCM)
17:0	RO/V	18'h0	**ADDR** This field captures the invalid QW Flash Address that was detected during a violation. Upper 2 address bits are assumed to be all 0's. Bit 2 is undefined. Lower 2 address bits are assumed to be all 0's.

12.4.2.14 MPR_VDATA (MPR_VDATA)

Memory Protection Region Violation Data Value

MEM Offset (B0100000)	34h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	CCCC_CCCCh

Bits	Access Type	Default	Description
31:0	RW/O	32'hCCCCCCCC	**MPR Violation Data (VDATA)** This field controls the data returned to Agents that violate the MPR Access rules.

12.4.3 Internal SRAM Register Summary

MEM BaseAddress: 0xB0400000

MEM Address	Default	Name	Description
0x0	0000_0000h	MPR0_CFG	MPR_CFG
0x4	0000_0000h	MPR1_CFG	MPR_CFG
0x8	0000_0000h	MPR2_CFG	MPR_CFG
0xC	0000_0000h	MPR3_CFG	MPR_CFG
0x10	FFFF_FFFFh	MPR_VDATA	MPR_VDATA
0x14	0000_0000h	MPR_VSTS	MPR_VSTS

12.4.4 Internal SRAM Register Detailed Description

12.4.4.1 MPR_CFG (MPR0_CFG)

Memory Protection Region Configuration Register

MEM Offset (B0400000)	0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'b0	**MPR Lock (LOCK)** Lock the Memory Protection Region Configuration.
30	RW/L	1'b0	**MPR Enable (ENABLE)** Enable the Memory Protection Region.
29:27	RO	3'b0	**RSV4** Reserved
26:24	RW/L	3'b0	**MPR Read Access Allow (RD_ALLOW)** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA
23	RO	1'b0	**RSV3** Reserved
22:20	RW/L	3'b0	**MPR Write Access Allow (WR_ALLOW)** Enable Write Access on an Agent by Agent basis: Bit [0] : Enables Write Access for the Host Processor Bit [1] : Reserved Bit [2] : Enables Write Access for DMA
19:17	RO	3'b0	**RSV2** Reserved
16:10	RW/L	7'b0	**MPR Upper Bound (UPR_BOUND)** The Upper Address Bound is compared with 16:10 of the incoming address determine the upper 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 1's. So, incoming address is checked for **less than or equal to** this field.
9:7	RO	3'b0	**RSV1** Reserved
6:0	RW/L	7'b0	**MPR Lower Bound (LWR_BOUND)** The Lower Address Bound is compared with 16:10 of the incoming address determine the lower 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 0's. So, incoming address is checked for **greater than or equal to** this field.

12.4.4.2 MPR_CFG (MPR1_CFG)

Memory Protection Region Configuration Register

MEM Offset (B0400000)	4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'b0	**MPR Lock (LOCK)** Lock the Memory Protection Region Configuration.
30	RW/L	1'b0	**MPR Enable (ENABLE)** Enable the Memory Protection Region.
29:27	RO	3'b0	**RSV4** Reserved
26:24	RW/L	3'b0	**MPR Read Access Allow (RD_ALLOW)** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA
23	RO	1'b0	**RSV3** Reserved
22:20	RW/L	3'b0	**MPR Write Access Allow (WR_ALLOW)** Enable Write Access on an Agent by Agent basis: Bit [0] : Enables Write Access for the Host Processor Bit [1] : Reserved Bit [2] : Enables Write Access for DMA
19:17	RO	3'b0	**RSV2** Reserved
16:10	RW/L	7'b0	**MPR Upper Bound (UPR_BOUND)** The Upper Address Bound is compared with 16:10 of the incoming address determine the upper 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 1's. So, incoming address is checked for **less than or equal to** this field.
9:7	RO	3'b0	**RSV1** Reserved

Bits	Access Type	Default	Description
6:0	RW/L	7'b0	**MPR Lower Bound (LWR_BOUND)** The Lower Address Bound is compared with 16:10 of the incoming address determine the lower 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 0's. So, incoming address is checked for **greater than or equal to** this field.

12.4.4.3 MPR_CFG (MPR2_CFG)

Memory Protection Region Configuration Register

MEM Offset (B0400000)	8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'b0	**MPR Lock (LOCK)** Lock the Memory Protection Region Configuration.
30	RW/L	1'b0	**MPR Enable (ENABLE)** Enable the Memory Protection Region.
29:27	RO	3'b0	**RSV4** Reserved
26:24	RW/L	3'b0	**MPR Read Access Allow (RD_ALLOW)** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA
23	RO	1'b0	**RSV3** Reserved
22:20	RW/L	3'b0	**MPR Write Access Allow (WR_ALLOW)** Enable Write Access on an Agent by Agent basis: Bit [0] : Enables Write Access for the Host Processor Bit [1] : Reserved Bit [2] : Enables Write Access for DMA
19:17	RO	3'b0	**RSV2** Reserved

Bits	Access Type	Default	Description
16:10	RW/L	7'b0	**MPR Upper Bound (UPR_BOUND)** The Upper Address Bound is compared with 16:10 of the incoming address determine the upper 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 1's. So, incoming address is checked for **less than or equal to** this field.
9:7	RO	3'b0	**RSV1** Reserved
6:0	RW/L	7'b0	**MPR Lower Bound (LWR_BOUND)** The Lower Address Bound is compared with 16:10 of the incoming address determine the lower 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 0's. So, incoming address is checked for **greater than or equal to** this field.

12.4.4.4 MPR_CFG (MPR3_CFG)

Memory Protection Region Configuration Register

MEM Offset (B0400000)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1S	1'b0	**MPR Lock (LOCK)** Lock the Memory Protection Region Configuration.
30	RW/L	1'b0	**MPR Enable (ENABLE)** Enable the Memory Protection Region.
29:27	RO	3'b0	**RSV4** Reserved
26:24	RW/L	3'b0	**MPR Read Access Allow (RD_ALLOW)** Enable Read Access on an Agent by Agent basis: [0] : Enables Read Access for the Host Processor [1] : Reserved [2] : Enables Read Access for DMA
23	RO	1'b0	**RSV3** Reserved

Bits	Access Type	Default	Description
22:20	RW/L	3'b0	**MPR Write Access Allow (WR_ALLOW)** Enable Write Access on an Agent by Agent basis: Bit [0] : Enables Write Access for the Host Processor Bit [1] : Reserved Bit [2] : Enables Write Access for DMA
19:17	RO	3'b0	**RSV2** Reserved
16:10	RW/L	7'b0	**MPR Upper Bound (UPR_BOUND)** The Upper Address Bound is compared with 16:10 of the incoming address determine the upper 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 1's. So, incoming address is checked for **less than or equal to** this field.
9:7	RO	3'b0	**RSV1** Reserved
6:0	RW/L	7'b0	**MPR Lower Bound (LWR_BOUND)** The Lower Address Bound is compared with 16:10 of the incoming address determine the lower 1KB aligned value of the Protected Region. Upper 4 bits are unused by HW. For address comparison purposes, lower address bits[9:0] of incoming address is assumed to be all 0's. So, incoming address is checked for **greater than or equal to** this field.

12.4.4.5 MPR_VDATA (MPR_VDATA)

Memory Protection Region Violation Data Value

MEM Offset (B0400000)	10h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	FFFF_FFFFh

Bits	Access Type	Default	Description
31:0	RW/O	32'hFFFFFFFF	**MPR Violation Data (VDATA)** This field controls the data returned to Agents that violate the MPR Access rules.

12.4.4.6 MPR_VSTS (MPR_VSTS)

Memory Protection Region Violation Details

MEM Offset (B0400000)	14h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31	RW/1C/V	1'b0	**MPR Violation Details Valid (VALID)** This field is asserted when a Violation Event Occurs. No further Violation Events will be captured in this register until this bit is cleared by SW.
30:20	RO	11'b0	**RSV** Reserved
19:18	RO/V	2'b0	**MPR Violation Agent ID (AGENT)** This field captures the Agent ID for an Access Violation: 0: Host Processor 1: Reserved 2: DMA
17	RO/V	1'b0	**MPR Violation Agent ID (TYPE)** This field captures the Transfer Type for an Access Violation: 0: Read 1: Write
16:0	RO/V	17'b0	**MPR Violation Address (ADDR)** This field captures the Address for an Access Violation. Upper 4b are assumed to be all 0's. Bit 2 is undefined. Lower 2 address bits are assumed to be all 0's.

13 I^{2}C

The SoC implements one instance of an I^{2}C controller, which can operate in master mode or slave mode as configured. Both 7 bit and 10 bit addressing modes are supported.

13.1 Signal Descriptions

Please see Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, "Electrical Characteristics"
- Description: A brief explanation of the signal's function

Table 35. Memory 0 Signals

Signal Name	Direction/ Type	Description
I2C_0_CLK	I/O	I^{2}C Serial Clock
I2C_0_DATA	I/O	I^{2}C Serial Data

13.2 Features

- One I^{2}C Interface
- Support both Master and Slave operation
- Operational Speeds:
  - Standard Mode (0 to 100 Kbps)
  - Fast Mode (≤ 400 Kbps)
  - Fast Mode Plus (≤ 1 Mbps)
- 7 bit or 10 bit Addressing
- Supports Clock Stretching by Slave Devices
- Multi-Master Arbitration
- Spike Suppression
- Hardware Handshake Interface to support DMA capability
- Interrupt Control
- FIFO support with 16B deep RX and TX FIFO's

13.3 Memory Mapped IO Registers

Registers listed are for I2C 0, starting at base address B0002800h.

Table 36. Summary of I^{2}C Registers—0xB0002800

MEM Address	Default	Instance Name	Name
0xB0002800	0000_007Fh	IC_CON	Control Register
0xB0002804	0000_2055h	IC_TAR	Master Target Address
0xB0002808	0000_0055h	IC_SAR	Slave Address
0xB000280C	0000_0001h	IC_HS_MADDR	High Speed Master ID
0xB0002810	0000_0000h	IC_DATA_CMD	Data Buffer and Command
0xB0002814	0000_0190h	IC_SS_SCL_HCNT	Standard Speed Clock SCL High Count
0xB0002818	0000_01D6h	IC_SS_SCL_LCNT	Standard Speed Clock SCL Low Count
0xB000281C	0000_003Ch	IC_FS_SCL_HCNT	Fast Speed Clock SCL High Count
0xB0002820	0000_0082h	IC_FS_SCL_LCNT	Fast Speed I^{2}C Clock SCL Low Count
0xB0002824	0000_0006h	IC_HS_SCL_HCNT	High Speed I^{2}C Clock SCL High Count
0xB0002828	0000_0010h	IC_HS_SCL_LCNT	High Speed I^{2}C Clock SCL Low Count
0xB000282C	0000_0000h	IC_INTR_STAT	Interrupt Status
0xB0002830	0000_18FFh	IC_INTR_MASK	Interrupt Mask
0xB0002834	0000_0000h	IC_RAW_INTR_STAT	Raw Interrupt Status
0xB0002838	0000_000Fh	IC_RX_TL	Receive FIFO Threshold Level
0xB000283C	0000_0000h	IC_TX_TL	Transmit FIFO Threshold Level
0xB0002840	0000_0000h	IC_CLR_INTR	Clear Combined and Individual Interrupt
0xB0002844	0000_0000h	IC_CLR_RX_UNDER	Clear RX_UNDER Interrupt
0xB0002848	0000_0000h	IC_CLR_RX_OVER	Clear RX_OVER Interrupt
0xB000284C	0000_0000h	IC_CLR_TX_OVER	Clear TX_OVER Interrupt
0xB0002850	0000_0000h	IC_CLR_RD_REQ	Clear RD_REQ Interrupt
0xB0002854	0000_0000h	IC_CLR_TX_ABRT	Clear TX_ABRT Interrupt
0xB0002858	0000_0000h	IC_CLR_RX_DONE	Clear RX_DONE Interrupt
0xB000285C	0000_0000h	IC_CLR_ACTIVITY	Clear ACTIVITY Interrupt
0xB0002860	0000_0000h	IC_CLR_STOP_DET	Clear STOP_DET Interrupt
0xB0002864	0000_0000h	IC_CLR_START_DET	Clear START_DET Interrupt
0xB0002868	0000_0000h	IC_CLR_GEN_CALL	Clear GEN_CALL Interrupt
0xB000286C	0000_0000h	IC_ENABLE	Enable
0xB0002870	0000_0006h	IC_STATUS	Status

MEM Address	Default	Instance Name	Name
0xB0002874	0000_0000h	IC_TXFLR	Transmit FIFO Level
0xB0002878	0000_0000h	IC_RXFLR	Receive FIFO Level
0xB000287C	0001_0001h	IC_SDA_HOLD	SDA Hold
0xB0002880	0000_0000h	IC_TX_ABRT_SOURCE	Transmit Abort Source
0xB0002888	0000_0000h	IC_DMA_CR	SDA Setup
0xB000288C	0000_0000h	IC_DMA_TDLR	DMA Transmit Data Level Register
0xB0002890	0000_0000h	IC_DMA_RDLR	I^{2}C Receive Data Level Register
0xB0002894	0000_0064h	IC_SDA_SETUP	SDA Setup
0xB0002898	0000_0001h	IC_ACK_GENERAL_CALL	General Call Ack
0xB000289C	0000_0000h	IC_ENABLE_STATUS	Enable Status
0xB00028A0	0000_0007h	IC_FS_SPKLEN	SS and FS Spike Suppression Limit
0xB00028A4	0000_0000h	IC_HS_SPKLEN	HS spike suppression limit
0xB00028A8	0000_0000h	IC_CLR_RESTART_DET	clear the RESTART_DET interrupt
0xB00028F4	000F_0FAEh	IC_COMP_PARAM_1	Configuration Parameters
0xB00028F8	3132_322Ah	IC_COMP_VERSION	Component Version
0xB00028FC	4457_0140h	IC_COMP_TYPE	Component Type

13.3.1.1 Control Register (IC_CON)

Used to control the I^{2}C controller. Can be written only when the I^{2}C is disabled (IC_ENABLE=0). Writes at other times have no effect.

MEM Offset ()	0B0002800h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_007Fh

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV)**
9	RW	1'b0	**RX_FIFO_FULL_HLD_CTRL (RX_FIFO_FULL_HLD_CTRL)** This bit controls whether the bus should be held when the Rx FIFO is physically full to its RX_BUFFER_DEPTH, as described in the IC_RX_FULL_HLD_BUS_EN parameter. Dependencies: This register bit value is applicable only when the IC_RX_FULL_HLD_BUS_EN configuration parameter is set to 1. If IC_RX_FULL_HLD_BUS_EN = 0, then this bit is read-only. If IC_RX_FULL_HLD_BUS_EN = 1, then this bit can be read or write.

Bits	Access Type	Default	Description
8	RW	1'b0	**TX_EMPTY_CTRL (TX_EMPTY_CTRL)** This bit controls the generation of the TX_EMPTY interrupt, as described in the IC_RAW_INTR_STAT register.
7	RW	1'b0	**STOP_DET_IFADDRESSED (STOP_DET_IFADDRESSED)** In slave mode: 1b1 issues the STOP_DET interrrupt only when it is addressed. 1b0 issues the STOP_DET irrespective of whether its addressed or not.
6	RW	1'b1	**Slave Mode Disable (IC_SLAVE_DISABLE)** This bit controls whether I^{2}C has its slave disabled. If this bit is set (slave disabled), I^{2}C controller functions only as a master. 0: slave is enabled 1: slave is disabled NOTE: Software must ensure slave and master mode are mutually exclusive. IMPORTANT: if IC_SLAVE_DISABLE == 0 --> MASTER_MODE == 0
5	RW	1'b1	**Restart Support (IC_RESTART_EN)** Determines whether RESTART conditions may be sent when acting as a master. Some older slaves do not support handling RESTART conditions; however, RESTART conditions are used in several I^{2}C controller operations. 0: disable 1: enable When RESTART is disabled, the master is prohibited from performing the following functions: - Change direction within a transfer (split) - Send a START BYTE - High-speed mode operation - Combined format transfers in 7-bit addressing modes - Read operation with a 10-bit address - Send multiple bytes per transfer By replacing RESTART condition followed by a STOP and a subsequent START condition, split operations are broken down into multiple I^{2}C transfers. If the above operations are performed, it will result in setting TX_ABRT of the IC_RAW_INTR_STAT register
4	RW	1'b1	**Master Addressing Mode (IC_10BITADDR_MASTER)** Controls whether the I^{2}C controller starts its transfers in 7- or 10-bit addressing mode when acting as a master. 0: 7-bit addressing 1: 10-bit addressing
3	RW	1'b1	**Slave Addressing Mode (IC_10BITADDR_SLAVE)** When acting as a slave, this bit controls whether the I^{2}C controller responds to 7- or 10-bit addresses. - 0: 7-bit addressing ignores transactions that involve 10-bit addressing; for 7-bit addressing, only the lower 7 bits of the IC_SAR register are compared. - 1: 10-bit addressing responds to only 10-bit addressing transfers that match the full 10 bits of the IC_SAR register.

Bits	Access Type	Default	Description
2:1	RW	2'b11	**Speed Mode (SPEED)** I^{2}C Master operational speed. Relevant only in master mode. Mode must be set by firmware. 01: standard mode (100 kbit/s) 10: fast mode (400 kbit/s)
0	RW	1'b1	**Master Mode Enable (MASTER_MODE)** This bit controls whether the I^{2}C master is enabled. 0: master disabled 1: master enabled NOTE: Software must ensure master and slave mode are mutually exclusive if MASTER_MODE == 1 --> IC_SLAVE_DISABLE == 1

13.3.1.2 Master Target Address (IC_TAR)

Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002804h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_2055h

Bits	Access Type	Default	Description	PowerWell	ResetSignal
31:13	RO	19'h00001	**Reserved (RSV).**		
12	RW	1'b0	**IC_10BITADDR_MASTER (IC_10BITADDR_MASTER)** This bit controls whether the I^{2}C controller starts its transfers in 7- or 10-bit addressing mode when acting as a master. 0: 7-bit addressing 1: 10-bit addressing		
11	RW	1'b0	**Special Command Enable (SPECIAL)** This bit indicates whether software performs a General Call or START BYTE command. 0: ignore bit 10 GC_OR_START and use IC_TAR normally 1: perform special I2C command as specified in GC_OR_START field		

Bits	Access Type	Default	Description	PowerWell	ResetSignal
10	RW	1'b0	**Special Command Type (GC_OR_START)** If IC_TAR bit 11 (SPECIAL) is set to 1, then this bit indicates whether a General Call or START byte command is to be performed by the I2C Controller. 0: General Call Address after issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TX_ABRT) of the IC_RAW_INTR_STAT register. I^{2}C controller remains in General Call mode until the SPECIAL bit value (bit 11) is cleared. 1: START BYTE		
9:0	RW	10'h055	**Master Target Address (IC_TAR)** This is the target address for any master transaction. When transmitting a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. If the IC_TAR and IC_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to only a slave. IMPORTANT: if MASTER_MODE == 1 --> IC_SLAVE_DISABLE == 1		

13.3.1.3 Slave Address (IC_SAR)

Holds the slave address when the I^{2}C is operating as a slave. Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect

MEM Offset ()	0B0002808h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0055h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV)**
9:0	RW	10'h055	**Slave Address (IC_SAR)** For 7-bit addressing, only IC_SAR[6:0] is used.

13.3.1.4 High Speed Master ID (IC_HS_MADDR)

I2C High Speed Master Mode Code Address. Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B000280Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved**
2:0	RW	3'b01	**HS Master Code (IC_HS_MAR)** This bit field holds the value of the I^{2}C HS mode master code. HS-mode master codes are reserved 8-bit codes (00001xxx) that are not used for slave addressing or other purposes. Each master has its unique master code; up to eight high-speed mode masters can be present on the same I2C bus system. Valid values are from 0 to 7.

13.3.1.5 Data Buffer and Command (IC_DATA_CMD)

CPU writes to it when filling the TX FIFO and the reads from when retrieving bytes from RX FIFO.

MEM Offset ()	0B0002810h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:11	RO	21'b0	**Reserved (RSV)**
10	RO	1'b0	**Restart Bit Control (RESTART)** This bit controls whether a RESTART is issued before the byte is sent or received. - 1 if IC_RESTART_EN is 1, a RESTART is issued before the data is sent/received (according to the value of CMD), regardless of whether or not the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead. - 0 If IC_RESTART_EN is 1, a RESTART is issued only if the transfer direction is changing from the previous command; if IC_RESTART_EN is 0, a STOP followed by a START is issued instead

Bits	Access Type	Default	Description
9	RO	1'b0	**Stop Bit Control (STOP)** This bit controls whether a STOP is issued after the byte is sent or received: - 1 STOP is issued after this byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. - 0 STOP is not issued after this byte, regardless of whether or not the Tx FIFO is empty. If the Tx FIFO is not empty, the master continues the current transfer by sending/receiving data bytes according to the value of the CMD bit. If the Tx FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the Tx FIFO.
8	RW	1'b0	**Command (CMD)** This bit controls whether a read or a write is performed. This bit controls the direction only in I^{2}C master mode. 0 = Write 1 = Read When a command is entered in the TX FIFO, this bit distinguishes the write and read commands. In slave-receiver mode, this bit is a 'don't care' because writes to this register are not required. In slave-transmitter mode, a '0' indicates that CPU data is to be transmitted and as DAT or IC_DATA_CMD[7:0]. NOTE: when programming this bit, attempting to perform a read operation after a General Call command has been sent results in a TX_ABRT interrupt (bit 6 of the IC_RAW_INTR_STAT register), unless bit 11 (SPECIAL) in the IC_TAR register has been cleared. If a '1' is written to this bit after receiving a RD_REQ interrupt, then a TX_ABRT interrupt occurs. NOTE: It is possible that while attempting a master I^{2}C read transfer, a RD_REQ interrupt may have occurred simultaneously due to a remote I^{2}C master addressing I^{2}C controller. In this type of scenario, the I^{2}C controller ignores the IC_DATA_CMD write, generates a TX_ABRT interrupt, and waits to service the RD_REQ interrupt.
7:0	RW	8'b0	**Data Buffer (DAT)** Contains the data to be transmitted or received on the I^{2}C bus. When writing to this register and want to perform a read, DAT field is ignored by the I^{2}C controller. When reading this register, DAT return the value of data received on the I^{2}C controller interface.

13.3.1.6 Standard Speed Clock SCL High Count (IC_SS_SCL_HCNT)

Sets the SCL clock high-period count for standard speed (SS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002814h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0190h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h0190	**SS SCL clock high-period count (IC_SS_SCL_HCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. This register must not be programmed to a value higher than 65525, because I^{2}C controller uses a 16-bit counter to flag an I^{2}C bus idle condition when this counter reaches a value of IC_SS_SCL_HCNT + 10.

13.3.1.7 Standard Speed Clock SCL Low Count (IC_SS_SCL_LCNT)

Sets the SCL clock low-period count for standard speed (SS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002818h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_01D6h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h01d6	**SS SCL clock low-period count (IC_SS_SCL_LCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set.

13.3.1.8 Fast Speed Clock SCL High Count (IC_FS_SCL_HCNT)

Sets the SCL clock high-period count for fast speed (FS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B000281Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_003Ch

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h003c	**FS SCL clock high-period count (IC_FS_SCL_HCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.

13.3.1.9 Fast Speed I^{2}C Clock SCL Low Count (IC_FS_SCL_LCNT)

Sets the SCL clock low-period count for fast speed (FS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002820h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0082h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h0082	**SS SCL clock low-period count (IC_FS_SCL_LCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set.

13.3.1.10 High Speed I^{2}C Clock SCL High Count (IC_HS_SCL_HCNT)

Sets the SCL clock high-period count for high speed (HS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002824h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0006h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h0006	**HS SCL clock high-period count (IC_HS_SCL_HCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. The SCL High time depends on the loading of the bus. For 100pF loading, the SCL High time is 60ns; for 400pF loading, the SCL High time is 120ns. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.

13.3.1.11 High Speed I^{2}C Clock SCL Low Count (IC_HS_SCL_LCNT)

Sets the SCL clock low-period count for high speed (HS). Can be written only when the I^{2}C is disabled (IC_ENABLE==0). Writes at other times have no effect.

MEM Offset ()	0B0002828h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0010h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved (RSV)**
15:0	RW	16'h0010	**SS SCL clock low-period count (IC_HS_SCL_LCNT)** Must be set before any I^{2}C bus transaction can take place to ensure proper I/O timing. For 100pF loading, the SCL High time is 60ns; for 400pF loading, the SCL High time is 120ns. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set.

13.3.1.12 Interrupt Status (IC_INTR_STAT)

Each bit in this register has a corresponding mask bit in the IC_INTR_MASK register. These bits are cleared by reading the matching interrupt clear register. The unmasked raw versions of these bits are available in the IC_RAW_INTR_STAT register.

MEM Offset ()	0B000282Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:12	RO	20'b0	**Reserved (RSV)**
11	RO	1'b0	**General Call Acknowledged (R_GEN_CALL)** Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling the I^{2}C controller or when the CPU reads bit 0 of the IC_CLR_GEN_CALL register. I^{2}C controller stores the received data in the Rx buffer.
10	RO	1'b0	**Start Detected (R_START_DET)** Indicates whether a START or RESTART condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
9	RO	1'b0	**Stop Detected (R_STOP_DET)** Indicates whether a STOP condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
8	RO	1'b0	**Activity (R_ACTIVITY)** This bit captures I^{2}C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus
7	RO	1'b0	**RX Completed (R_RX_DONE)** When the I^{2}C controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.
6	RO	1'b0	**TX Abort (R_TX_ABRT)** This bit indicates if the I^{2}C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I^{2}C master or an I^{2}C slave, and is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission.

Bits	Access Type	Default	Description
5	RO	1'b0	**Read Requested (R_RD_REQ)** This bit is set to 1 when I I^{2}C 2C controller is acting as a slave and another I^{2}C master is attempting to read data from it. The controller holds the I^{2}C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the IC_DATA_CMD register. This bit is set to 0 just after the processor reads the IC_CLR_RD_REQ register.
4	RO	1'b0	**TX Empty (R_TX_EMPTY)** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with ic_en=0, this bit is set to 0. Reset value.
3	RO	1'b0	**TX Overflow (R_TX_OVER)** Set during transmit if the transmit buffer is filled to IC_TX_BUFFER_DEPTH and the processor attempts to issue another I^{2}C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.
2	RO	1'b0	**RX Full (R_RX_FULL)** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. This bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.
1	RO	1'b0	**RX Overflow (R_RX_OVER)** Set if the receive buffer is completely filled to IC_RX_BUFFER_DEPTH and an additional byte is received from an external I^{2}C device. The I^{2}C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.
0	RO	1'b0	**RX Underflow (R_RX_UNDER)** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.

13.3.1.13 Interrupt Mask (IC_INTR_MASK)

These bits mask their corresponding interrupt status bits. They are active high; a value of 0 prevents a bit from generating an interrupt.

MEM Offset ()	0B0002830h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_18FFh

Bits	Access Type	Default	Description
31:13	RO	19'b0	**Reserved (RSV)**
12	RW	1'b1	**M_RESTART_DET Mask (M_RESTART_DET)** This bit masks the R_RESTART_DET interrupt status bit in the IC_INTR_STAT register.
11	RW	1'b1	**General Call Acknowledged Mask (M_GEN_CALL)** Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling the I^{2}C controller or when the CPU reads bit 0 of the IC_CLR_GEN_CALL register. I^{2}C controller stores the received data in the Rx buffer.
10	RW	1'b0	**Start Detected Mask (M_START_DET)** Indicates whether a START or RESTART condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
9	RW	1'b0	**Stop Detected Mask (M_STOP_DET)** Indicates whether a STOP condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
8	RW	1'b0	**Activity Mask (M_ACTIVITY)** This bit captures I^{2}C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus.
7	RW	1'b1	**RX Completed Mask (M_RX_DONE)** When the I^{2}C controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.

Bits	Access Type	Default	Description
6	RW	1'b1	**TX Abort Mask (M_TX_ABRT)** This bit indicates if the I^{2}C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I^{2}C master or an I^{2}C slave, and is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission.
5	RW	1'b1	**Read Requested Mask (M_RD_REQ)** This bit is set to 1 when I^{2}C controller is acting as a slave and another I^{2}C master is attempting to read data from it. The controller holds the I^{2}C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the IC_DATA_CMD register. This bit is set to 0 just after the processor reads the IC_CLR_RD_REQ register.
4	RW	1'b1	**TX Empty Mask (M_TX_EMPTY)** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with IC_EN=0, this bit is set to 0. Reset value.
3	RW	1'b1	**TX Overflow Mask (M_TX_OVER)** Set during transmit if the transmit buffer is filled to IC_TX_BUFFER_DEPTH and the processor attempts to issue another I^{2}C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.
2	RW	1'b1	**RX Full Mask (M_RX_FULL)** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. This bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.
1	RW	1'b1	**RX Overflow Mask (M_RX_OVER)** Set if the receive buffer is completely filled to IC_RX_BUFFER_DEPTH and an additional byte is received from an external I^{2}C device. The I^{2}C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.

Bits	Access Type	Default	Description
0	RW	1'b1	**RX Underflow Mask (M_RX_UNDER)** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.

13.3.1.14 Raw Interrupt Status (IC_RAW_INTR_STAT)

Unlike the IC_INTR_STAT register, these bits are not masked so they always show the true status of the I^{2}C controller.

MEM Offset ()	0B0002834h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:13	RO	19'b0	**Reserved (RSV)**
12	RO	1'b0	**RESTART condition has occurred (RESTART_DET)** Indicates whether a RESTART condition has occurred on the I^{2}C interface when I^{2}C controller is operating in slave mode and the slave is the addressed slave.
11	RO	1'b0	**General Call Acknowledged (GEN_CALL)** Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling the I^{2}C controller or when the CPU reads bit 0 of the IC_CLR_GEN_CALL register. I^{2}C controller stores the received data in the Rx buffer.
10	RO	1'b0	**Start Detected (START_DET)** Indicates whether a START or RESTART condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
9	RO	1'b0	**Stop Detected (STOP_DET)** Indicates whether a STOP condition has occurred on the I^{2}C interface regardless of whether the controller is operating in slave or master mode.
8	RO	1'b0	**Activity (ACTIVITY)** This bit captures I^{2}C controller activity and stays set until it is cleared. There are four ways to clear it: - Disabling the controller - Reading the IC_CLR_ACTIVITY register - Reading the IC_CLR_INTR register - System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller is idle, this bit remains set until cleared, indicating that there was activity on the bus.

Bits	Access Type	Default	Description
7	RO	1'b0	**RX Completed (RX_DONE)** When the I^{2}C controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.
6	RO	1'b0	**TX Abort (TX_ABRT)** This bit indicates if the I^{2}C controller, in transmitter mode, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I^{2}C master or an I^{2}C slave, and is referred to as a 'transmit abort'. When this bit is set to 1, the IC_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register IC_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes for transmission.
5	RO	1'b0	**Read Requested (RD_REQ)** This bit is set to 1 when I^{2}C controller is acting as a slave and another I^{2}C master is attempting to read data from it. The controller holds the I^{2}C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the IC_DATA_CMD register. This bit is set to 0 just after the processor reads the IC_CLR_RD_REQ register.
4	RO	1'b0	**TX Empty (TX_EMPTY)** This bit is set to 1 when the transmit buffer is at or below the threshold value set in the IC_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with IC_EN=0, this bit is set to 0. Reset value.
3	RO	1'b0	**TX Overflow (TX_OVER)** Set during transmit if the transmit buffer is filled to IC_TX_BUFFER_DEPTH and the processor attempts to issue another I^{2}C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.
2	RO	1'b0	**RX Full (RX_FULL)** Set when the receive buffer reaches or goes above the RX_TL threshold in the IC_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (IC_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the IC_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.
1	RO	1'b0	**RX Overflow (RX_OVER)** Set if the receive buffer is completely filled to IC_RX_BUFFER_DEPTH and an additional byte is received from an external I^{2}C device. The I^{2}C Controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.

Bits	Access Type	Default	Description
0	RO	1'b0	**RX Underflow (RX_UNDER)** Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (IC_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when IC_EN goes to 0, this interrupt is cleared.

13.3.1.15 Receive FIFO Threshold Level (IC_RX_TL)

Controls the level of entries (or above) that triggers the RX_FULL interrupt (bit 2 in IC_RAW_INTR_STAT register).

MEM Offset ()	0B0002838h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_000Fh

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved (RSV)**
3:0	RW	4'hF	**Receive FIFO Threshold Level (RX_TL)** The valid range is 0-255, with the additional restriction that hardware does not allow this value to be set to a value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 1 entry, and a value of 255 sets the threshold for 256 entries.

13.3.1.16 Transmit FIFO Threshold Level (IC_TX_TL)

Controls the level of entries (or below) that trigger the TX_EMPTY interrupt (bit 4 in IC_RAW_INTR_STAT register).

MEM Offset ()	0B000283Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved (RSV)**
3:0	RW	4'h0	**Transmit FIFO Threshold Level (TX_TL)** The valid range is 0-255, with the additional restriction that it may not be set to value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 0 entries, and a value of 255 sets the threshold for 255 entries.

13.3.1.17 Clear Combined and Individual Interrupt (IC_CLR_INTR)

Read this register to clear the combined interrupt, all individual interrupts, and the IC_TX_ABRT_SOURCE register.

MEM Offset ()	0B0002840h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear Combined and Individual Interrupt (CLR_INTR)** This bit does not clear hardware clearable interrupts but software clearable interrupts. Refer to Bit 9 of the IC_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE.

13.3.1.18 Clear RX_UNDER Interrupt (IC_CLR_RX_UNDER)

Clear a single interrupt type.

MEM Offset ()	0B0002844h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear RX_UNDER (CLR_RX_UNDER)** Read this register to clear the RX_UNDER interrupt (bit 0) of the IC_RAW_INTR_STAT.

13.3.1.19 Clear RX_OVER Interrupt (IC_CLR_RX_OVER)

Clear a single interrupt type.

MEM Offset ()	0B0002848h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear RX_OVER (CLR_RX_OVER)** Read this register to clear the RX_OVER interrupt (bit 1) of the IC_RAW_INTR_STAT.

13.3.1.20 Clear TX_OVER Interrupt (IC_CLR_TX_OVER)

Clear a single interrupt type.

MEM Offset ()	0B000284Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear TX_OVER (CLR_TX_OVER)** Read this register to clear the TX_OVER interrupt (bit 3) of the IC_RAW_INTR_STAT.

13.3.1.21 Clear RD_REQ Interrupt (IC_CLR_RD_REQ)

Clear a single interrupt type.

MEM Offset ()	0B0002850h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear RD_REQ (CLR_RD_REQ)** Read this register to clear the RD_REQ interrupt (bit 5) of the IC_RAW_INTR_STAT.

13.3.1.22 Clear TX_ABRT Interrupt (IC_CLR_TX_ABRT)

Clear a single interrupt type.

MEM Offset ()	0B0002854h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear TX_ABRT (CLR_TX_ABRT)** Read this register to clear the TX_ABRT interrupt (bit 6) of the IC_RAW_INTR_STAT register, and the IC_TX_ABRT_SOURCE register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. Refer to Bit 9 of the IC_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE.

13.3.1.23 Clear RX_DONE Interrupt (IC_CLR_RX_DONE)

Clear a single interrupt type.

MEM Offset ()	0B0002858h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear RX_DONE (CLR_RX_DONE)** Read this register to clear the RX_DONE interrupt (bit 7) of the IC_RAW_INTR_STAT.

13.3.1.24 Clear ACTIVITY Interrupt (IC_CLR_ACTIVITY)

Clear a single interrupt type.

MEM Offset ()	0B000285Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear ACTIVITY (CLR_TX_ABRT)** Reading this register clears the ACTIVITY interrupt if the I^{2}C is not active anymore. If the I^{2}C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT.

13.3.1.25 Clear STOP_DET Interrupt (IC_CLR_STOP_DET)

Clear a single interrupt type.

MEM Offset ()	0B0002860h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear STOP_DET (CLR_STOP_DET)** Read this register to clear the STOP_DET interrupt (bit 9) of the IC_RAW_INTR_STAT.

13.3.1.26 Clear START_DET Interrupt (IC_CLR_START_DET)

Clear a single interrupt type.

MEM Offset ()	0B0002864h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear START_DET (CLR_START_DET)** Read this register to clear the START_DET interrupt (bit 10) of the IC_RAW_INTR_STAT.

13.3.1.27 Clear GEN_CALL Interrupt (IC_CLR_GEN_CALL)

Clear a single interrupt type.

MEM Offset ()	0B0002868h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear GEN_CALL (CLR_GEN_CALL)** Read this register to clear the GEN_CALL interrupt (bit 11) of IC_RAW_INTR_STAT.

13.3.1.28 Enable (IC_ENABLE)

Controls whether the I^2C controller is enabled. Software can disable I^2C controller while it is active.

MEM Offset ()	0B000286Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1	RW	1'b0	**Abort I^2C Controller (ABORT)** The software can abort the I^2C transfer in master mode by setting this bit. The software can set this bit only when ENABLE is already set; otherwise, the controller ignores any write to ABORT bit. The software cannot clear the ABORT bit once set. In response to an ABORT, the controller issues a STOP and flushes the Tx FIFO after completing the current transfer, then sets the TX_ABORT interrupt after the abort operation. The ABORT bit is cleared automatically after the abort operation. When set, the controller initiates the transfer abort. 0: ABORT not initiated or ABORT done 1: ABORT operation in progress

Bits	Access Type	Default	Description
0	RW	1'b0	**Enable I^{2}C Controller (ENABLE)** 0: Disabled (TX/RX FIFOs are held in an erased state) 1: Enabled NOTE: ensure that the controller is disabled properly. When disabled, the following occurs: - The TX FIFO and RX FIFO get flushed. - Status bits in the IC_INTR_STAT register are still active until the I^{2}C Controller goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. There is a two I^{2}C clocks delay when enabling or disabling the controller.

13.3.1.29 Status (IC_STATUS)

Read-only register used to indicate the current transfer status and FIFO status. The status register may be read at any time. None of the bits in this register request an interrupt. When the I^{2}C is disabled by writing 0 in bit 0 of the IC_ENABLE register: bits 1 and 2 are set to 1, bits 3 and 4 are set to 0. When the master or slave state machines goes to idle and IC_EN=0: bits 5 and 6 are set to 0.

MEM Offset ()	0B0002870h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0006h

Bits	Access Type	Default	Description
31:7	RO	25'b0	**Reserved (RSV)**
6	RO	1'b0	**SLV_ACTIVITY (SLV_ACTIVITY)** When the Slave Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0: Slave FSM is in IDLE state so the Slave part is not Active 1: Slave FSM is not in IDLE state so the Slave part is Active
5	RO	1'b0	**Master FSM Activity Status (MST_ACTIVITY)** When the Master Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0: Master FSM is in IDLE state so the Master part is not Active 1: Master FSM is not in IDLE state so the Master part is Active NOTE: IC_STATUS[0]-that is, ACTIVITY bit-is the OR of SLV_ACTIVITY and MST_ACTIVITY bits.

Bits	Access Type	Default	Description
4	RO	1'b0	**Receive FIFO Completely Full (RFF)** When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0: Receive FIFO is not full 1: Receive FIFO is full
3	RO	1'b0	**Receive FIFO Not Empty (RFNE)** This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0: Receive FIFO is empty 1: Receive FIFO is not empty
2	RO	1'b1	**Transmit FIFO Completely Empty (TFE)** When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty
1	RO	1'b1	**Transmit FIFO Not Full (TFNF)** Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0: Transmit FIFO is full 1: Transmit FIFO is not full
0	RO	1'b0	**Activity (ACTIVITY)** Receive FIFO Not Empty. This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0: Receive FIFO is empty 1: Receive FIFO is not empty

13.3.1.30 Transmit FIFO Level (IC_TXFLR)

Contains the number of valid data entries in the transmit FIFO buffer.
It is cleared whenever: The I^{2}C is disabled, there is a transmit abort
(i.e. TX_ABRT bit is set in the IC_RAW_INTR_STAT register) or the slave bulk transmit mode is aborted.
The register increments whenever data is placed into the transmit FIFO and decrements when data is taken from the transmit FIFO.

MEM Offset ()	0B0002874h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**

Bits	Access Type	Default	Description
4:0	RO	5'b0	**Transmit FIFO Level (TXFLR)** Contains the number of valid data entries in the transmit FIFO.

13.3.1.31 Receive FIFO Level (IC_RXFLR)

This register contains the number of valid data entries in the receive FIFO buffer.
It is cleared whenever: The I^{2}C is disabled, whenever there is a transmit abort caused by any of the events tracked in IC_TX_ABRT_SOURCE.
The register increments whenever data is placed into the receive FIFO and decrements when data is taken from the receive FIFO.

MEM Offset ()	0B0002878h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**
4:0	RO	5'b0	**Receive FIFO Level (RXFLR)** Contains the number of valid data entries in the receive FIFO.

13.3.1.32 SDA Hold (IC_SDA_HOLD)

This register controls the amount of hold time on the SDA signal after a negative edge of SCL line in units of I^{2}C clock period. The value programmed must be greater than the minimum hold time in each mode for the value to be implemented: 1 cycle in master, 7 cycles in slave mode. Writes to this register succeed only when I^{2}C controller is disabled (IC_ENABLE=0).

MEM Offset ()	0B000287Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:24	RO	8'b0	**Reserved (RSV)**
23:16	RW	8'h1	**SDA Hold (IC_SDA_RX_HOLD)** Sets the required SDA hold time in units of IC_CLK period, when I^{2}C controller acts as a transmitter.
15:0	RW	16'h1	**IC_SDA_TX_HOLD (IC_SDA_TX_HOLD)** Sets the required SDA hold time in units of IC_CLK period, when I^{2}C controller acts as a receiver.

13.3.1.33 Transmit Abort Source (IC_TX_ABRT_SOURCE)

Used to indicate the source of the TX_ABRT interrupt. Except for Bit 9, this register is cleared whenever the IC_CLR_TX_ABRT register or the IC_CLR_INTR register is read. To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; RESTART must be enabled (IC_CON[5]=1), the SPECIAL bit must be cleared (IC_TAR[11]), or the GC_OR_START bit must be cleared (IC_TAR[10]). Once the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, Bit 9 clears for one cycle and is then re-asserted.

MEM Offset ()	0B0002880h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:23	RO	9'b0	**Reserved (TX_FLUSH_CNT)**
22:17	RO	6'b0	**Reserved (RSV)**
16	RO	1'b0	**ABRT_USER_ABRT (ABRT_USER_ABRT)** This is a master-mode-only bit. Master has detected the transfer abort (IC_ENABLE[1]).
15	RO	1'b0	**Slave Read Completion (ABRT_SLVRD_INTX)** Set if the processor side responds to a slave mode request for data to be transmitted to a remote master and user writes a 1 in CMD (bit 8) of IC_DATA_CMD register.
14	RO	1'b0	**Slave Lost Bus (ABRT_SLV_ARBLOST)** Set if slave lost the bus while transmitting data to a remote master. IC_TX_ABRT_SOURCE[12] is set at the same time. Note: Even though the slave never 'owns' the bus, something could go wrong on the bus. This is a fail safe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then I^{2}C controller no longer own the bus.
13	RO	1'b0	**Slave Flush TX FIFO (ABRT_SLVFLUSH_TXFIFO)** Set if slave has received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO.
12	RO	1'b0	**Master Lost Arbitration (ARB_LOST)** Set if master has lost arbitration, or if IC_TX_ABRT_SOURCE[14] is also set, then the slave transmitter has lost arbitration. Note: I^{2}C can be both master and slave at the same time.
11	RO	1'b0	**Master Disabled (ABRT_MASTER_DIS)** Set if user tries to initiate a Master operation with the Master mode disabled.
10	RO	1'b0	**10 Bit Address READ and RESTART Disabled (ABRT_10B_RD_NORSTRT)** Set if the restart is disabled (IC_RESTART_EN bit (IC_CON[5]) =0) and the master sends a read command in 10-bit addressing mode.

Bits	Access Type	Default	Description
9	RO	1'b0	**START With RESTART Disabled (ABRT_SBYTE_NORSTRT)** To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; restart must be enabled (IC_CON[5]=1), the SPECIAL bit must be cleared (IC_TAR[11]), or the GC_OR_START bit must be cleared (IC_TAR[10]). Once the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets reasserted. Set if the restart is disabled (IC_RESTART_EN bit (IC_CON[5]) =0) and the user is trying to send a START Byte.
8	RO	1'b0	**HS Mode With RESTART Disabled (ABRT_HS_NORSTRT)** Set if the restart is disabled (IC_RESTART_EN bit (IC_CON[5]) =0) and the user is trying to use the master to transfer data in High Speed mode.
7	RO	1'b0	**START Acknowledged (ABRT_SBYTE_ACKDET)** Set if master has sent a START Byte and the START Byte was acknowledged (wrong behavior).
6	RO	1'b0	**HS Master ID Acknowledged (ABRT_HS_ACKDET)** Set if master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior).
5	RO	1'b0	**Read After General Call (ABRT_GCALL_READ)** Set if master sent a General Call but the user programmed the byte following the General Call to be a read from the bus (IC_DATA_CMD[9] is set to 1).
4	RO	1'b0	**General Call Not Acknowledged (ABRT_GCALL_NOACK)** 1: I^{2}C Controller in master mode sent a General Call and no slave on the bus acknowledged the General Call.
3	RO	1'b0	**TX Data Not Acknowledged (ABRT_TXDATA_NOACK)** Set if master has received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s).
2	RO	1'b0	**10 Bit Address Second Not Acknowledged (ABRT_10ADDR2_NOACK)** Set if master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave.
1	RO	1'b0	**10 Bit Address First Not Acknowledged (ABRT_10ADDR1_NOACK)** Set if master is in 10-bit address mode and the first 10-bit address byte was not acknowledged by any slave.
0	RO	1'b0	**7 Bit Address Not Acknowledged (ABRT_7B_ADDR_NOACK)** Set if master is in 7-bit addressing mode and the address sent was not acknowledged by any slave.

13.3.1.34 SDA Setup (IC_DMA_CR)

The register is used to enable the DMA Controller interface operation. There is a separate bit for transmit and receive. This can be programmed regardless of the state of IC_ENABLE.

MEM Offset ()	0B0002888h
Security_PolicyGroup	

IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1	RW	1'h0	**Transmit DMA Enable. (TDMAE)** This bit enables/disables the transmit FIFO DMA channel. 0b: Transmit DMA disabled 1b: Transmit DMA enabled
0	RW	1'b0	**Receive DMA Enable (RDMAE)** This bit enables/disables the receive FIFO DMA channel. 0b: Receive DMA disabled 1b: Receive DMA enabled

13.3.1.35 DMA Transmit Data Level Register (IC_DMA_TDLR)

MEM Offset () 0B000288Ch
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**
4:0	RW	5'h0	**Transmit Data Level (DMATDL)** Receive Data Level. This bit field controls the level at which a DMA request is made by the receive logic. It is equal to the watermark level, signal generated when the number of valid data entries in the transmit FIFO is equal to or below this field value, and TDMAE = 1.

13.3.1.36 I^{2}C Receive Data Level Register (IC_DMA_RDLR)

MEM Offset ()	0B0002890h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**
4:0	RW	5'h0	**Receive Data Level (DMARDL)** This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1, signal is generated when the number of valid data entries in the receive FIFO is equal to or more than this field value + 1, and RDMAE = 1.

13.3.1.37 SDA Setup (IC_SDA_SETUP)

Controls the amount of time delay (in terms of number of I^{2}C clock periods) introduced in the rising edge of SCL relative to SDA changing, by holding SCL low when servicing a read request while operating as a slave-transmitter. This register must be programmed with a value equal to or greater than 2.

MEM Offset ()	0B0002894h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0064h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7:0	RW	8'h64	**SDA Setup (SDA_SETUP)** It is recommended that if the required delay is 1us, then for an I^{2}C clock frequency of 10 MHz, IC_SDA_SETUP should be programmed to a value of 11.

13.3.1.38 General Call Ack (IC_ACK_GENERAL_CALL)

Controls whether the I^{2}C controller responds with a ACK or NACK when it receives an I^{2}C General Call address.

MEM Offset ()	0B0002898h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'h1	**I^{2}C General Call Ack (ACK_GEN_CALL)** When set to 1, the I^{2}C controller responds with a ACK when it receives a General Call. Otherwise, the controller responds with a NACK.

13.3.1.39 Enable Status (IC_ENABLE_STATUS)

Report the I^{2}C hardware status.

MEM Offset ()	0B000289Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved (RSV)**
2	RO	1'b0	**Slave Received Data Lost (SLV_RX_DATA_LOST)** Indicates if a Slave-Receiver operation has been aborted with at least one data byte received from an I^{2}C transfer due to the setting of IC_ENABLE from 1 to 0. When read as 1, the I^{2}C controller is deemed to have been actively engaged in an aborted I^{2}C transfer (with matching address) and the data phase of the I^{2}C transfer has been entered, even though a data byte has been responded with a NACK. NOTE: If the remote I^{2}C master terminates the transfer with a STOP condition before the controller has a chance to NACK a transfer, and IC_ENABLE has been set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to have been disabled without being actively involved in the data phase of a Slave-Receiver transfer. NOTE: The CPU can safely read this bit when IC_EN (bit 0) is read as 0.
1	RO	1'b0	**Slave Disabled While Busy (SLV_DISABLED_WHILE_BUSY)** This bit indicates if a potential or active Slave operation has been aborted due to the setting of the IC_ENABLE register from 1 to 0. This bit is set when the CPU writes a 0 to the IC_ENABLE register while: (a) the I^{2}C controller is receiving the address byte of the Slave-Transmitter operation from a remote master; OR, (b) address and data bytes of the Slave-Receiver operation from a remote master. When read as 1, the controller is deemed to have forced a NACK during any part of an I^{2}C transfer, irrespective of whether the I^{2}C address matches the slave address (IC_SAR register) OR if the transfer is completed before IC_ENABLE is set to 0 but has not taken effect. NOTE: If the remote I^{2}C master terminates the transfer with a STOP condition before the controller has a chance to NACK a transfer, and IC_ENABLE has been set to 0, then this bit will also be set to 1. When read as 0, controller is deemed to have been disabled when there is master activity, or when the I^{2}C bus is idle. NOTE: The CPU can safely read this bit when IC_EN (bit 0) is read as 0.

Bits	Access Type	Default	Description
0	RO	1'b0	**I^{2}C Enable Status (IC_EN)** When read as 1, the controller is deemed to be in an enabled state. When read as 0, the controller is deemed completely inactive. NOTE: The CPU can safely read this bit anytime. When this bit is read as 0, the CPU can safely read SLV_RX_DATA_LOST (bit 2) and SLV_DISABLED_WHILE_BUSY (bit 1).

13.3.1.40 SS and FS Spike Suppression Limit (IC_FS_SPKLEN)

Used to store the duration, measured in I^{2}C clock cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in Standard/Fast Speed modes. The relevant I^{2}C requirement is detailed in the I^{2}C Bus Specification. This register must be programmed with a minimum value of 2.

MEM Offset ()	0B00028A0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0007h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7:0	RW	8'h07	**I^{2}C SS and FS Spike Length (IC_FS_SPKLENRX_TL)** Must be set before any I^{2}C bus transaction can take place to ensure stable operation.

13.3.1.41 HS spike suppression limit (IC_HS_SPKLEN)

Used to store the duration, measured in I^{2}C clock cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in High Speed mode. This register must be programmed with a minimum value of 2.

MEM Offset ()	0B00028A4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h000000	**RSVD (RSVD)** Reserved
7:0	RW	8'b0	**Reserved (IC_HS_SPKLEN)**

13.3.1.42 Clear the RESTART_DET interrupt (IC_CLR_RESTART_DET)

Read this register to clear the RESTART_DET interrupt.

MEM Offset ()	0B00028A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RO	1'b0	**Clear RESTART_DET (CLR_RESTART_DET)** Read this register to clear the RESTART_DET interrupt.

13.3.1.43 Configuration Parameters (IC_COMP_PARAM_1)

Contains encoded information about the component's parameter settings.

Register Offset	0B00028F4h
IntelRsvd	True
Size	32 bits
Default	000F_0FAEh
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	000F_0FAEh	**RSVD (RSVD)** Reserved.

13.3.1.44 Component Version (IC_COMP_VERSION)

Register Offset	0B00028F8h
IntelRsvd	True
Size	32 bits
Default	3132_322Ah
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	3132_322Ah	**RSVD (RSVD)** Reserved.

13.3.1.45 Component Type (IC_COMP_TYPE)

Register Offset	0B00028FCh
IntelRsvd	True
Size	32 bits
Default	4457_0140h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	4457_0140h	**RSVD (RSVD)** Reserved.

§

14 UART

The SoC implements two instances of a 16550 compliant UART controller that supports baud rates between 300 baud and 2M baud. Hardware flow control is also supported. Both RS232 and RS485 are supported. 9-bit mode is also supported.

14.1 Signal Descriptions

Table 37. UART A Signals

Signal Name	Direction/ Type	Description
UART_A_TXD	Logic output	UART A single-ended Transmit data (RS232 or RS485). In RS485 mode, differential driver is outside SoC.
UART_A_RXD	Logic input	UART A single-ended Receive data (RS232 or RS485). In RS485 mode, differential receiver is outside SoC.
UART_A_RTS	Logic output	UART A Request To Send (RS232)
UART_A_CTS	Logic input	UART A Clear To Send (RS232)
UART_A_DE	Logic Output	UART A Driver Enable (RS485 mode). Used to control the differential driver of RS485 in platform/board. Polarity is configurable. This is multiplexed onto UART_A_RTS pin depending on RS485 or RS232 mode of operation.
UART_A_RE	Logic Output	UART B Receiver Enable (RS485 mode). Used to control the differential receiver of RS485 in platform/board. Polarity is configurable. This is multiplexed onto UART_B_CTS pin depending on RS485 or RS232 mode of operation.

Table 38. UART B Signals

Signal Name	Direction/ Type	Description
UART_B_TXD	Logic output	UART B single-ended Transmit data (RS232 or RS485). In RS485 mode, differential driver is outside SoC.
UART_B_RXD	Logic input	UART B single-ended Receive data (RS232 or RS485). In RS485 mode, differential receiver is outside SoC.
UART_B_RTS	Logic output	UART B Request To Send (RS232)
UART_B_CTS	Logic input	UART B Clear To Send (RS232)
UART_B_DE	Logic Output	UART B Driver Enable (RS485 mode). Used to control the differential driver of RS485 in platform/board. This is multiplexed onto UART_B_RTS pin depending on RS485 or RS232 mode of operation.
UART_B_RE	Logic Output	UART A Receiver Enable (RS485 mode). Used to control the differential receiver of RS485 in platform/board. This is multiplexed onto UART_B_CTS pin depending on RS485 or RS232 mode of operation.

14.2 Features

Both UART instances are configured identically, the following is a list of the UART controller features:

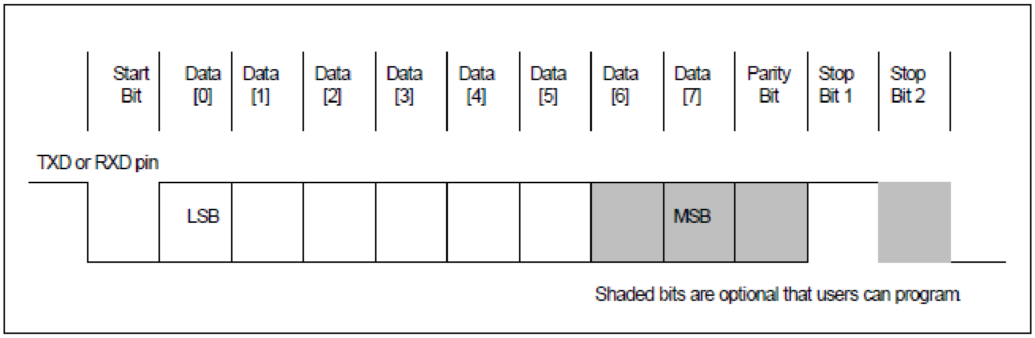
- Operation compliant with the 16550 Standard
  - Start bit
  - 5 to 9 bits of Data
  - Optional Parity bit (Odd or Even)
  - 1, 1.5 or 2 Stop bits
- Baud Rate configurability between 300 baud and 2M baud.
  - Maximum baud rate is limited by system clock frequency divided by 16.
  - Supported baud rates: 300, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 57600, 76800, 115200; multiples of 38.4kbps and multiples of 115.2kbps upto 2M baud
- Auto Flow Control mode as specified in the 16750 Standard
- Hardware Flow Control
- Software Flow Control (when Hardware Flow Control is disabled)
- Hardware Handshake Interface to support DMA capability
- Interrupt Control
- FIFO support with 16B TX and RX FIFO's
- Support of RS485
  - Differential driver/receiver is external to SoC.
  - Driver enable (DE) and Receiver enable (RE) outputs are driven from SoC to control the differential driver/receiver.
- Fractional clock divider that ensures less than 2% frequency error for most supported baud rates.
  - Fraction resolution is 4-bits.
  - Exception: 2.07% error for 1.391 Mbaud, 2.12% for 1.882 Mbaud and 2Mbaud, 2.53% error for 1.684 Mbaud.
- 9-bit data transfer mode to support multi-drop system where one master is connected to multiple slaves in a system

14.2.1 UART Function

The UART transmits and receives data in bit frames as shown below.

- Each data frame is between 7 and 12 bits long, depending on the size of data programmed and if parity and stop bits are enabled.
- The frame begins with a start bit that is represented by a high-to-low transition.
- Next, 5 to 8 bits of data is 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-one-half, or two stop bits (as programmed by users), which is represented by one or two successive bit periods of a logic one.

Figure 17. UART Data Format



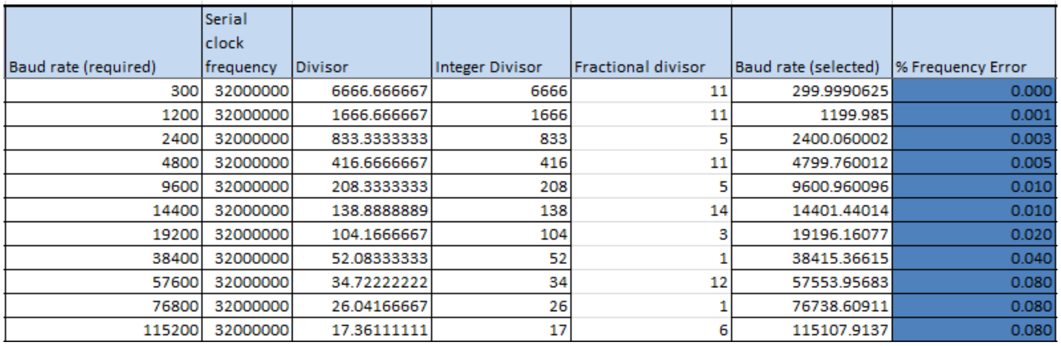
Each UART has a Transmit FIFO and a Receive FIFO and each holds 16 characters of data. There are two separate methods for moving data into/out of the FIFOs Interrupts and Polling.

14.2.2 Baud Rate Generator

The baud rates for the UARTs are generated from the base frequency (Fbase = Frequency of System Clock connected to UART Block) indicated in table below by programming the DLH and DLL registers as divisor. The hexadecimal value of the divisor is (IER_DLH[7:0]<<8) | RBR_THR_DLL[7:0]. The output baud rate is equal to the base frequency divided by sixteen times the value of the divisor, as follows: baud rate = (Fbase) / (16 * divisor). The UART also supports fractional dividers of 1/16 of the divisor by programming the DLF[3:0] register.

baud rate = (Fbase) / (16 * (divisor+(fdivisor/16))

Table 39. Baud Rate Calculator



Baud rate (required)	Serial clock frequency	Divisor	Integer Divisor	Fractional divisor	Baud rate (selected)	% Frequency Error
300	32000000	6666.666667	6666	11	299.9990625	0.000
1200	32000000	1666.666667	1666	11	1199.985	0.001
2400	32000000	833.3333333	833	5	2400.060002	0.003
4800	32000000	416.6666667	416	11	4799.760012	0.005
9600	32000000	208.3333333	208	5	9600.960096	0.010
14400	32000000	138.8888889	138	14	14401.44014	0.010
19200	32000000	104.1666667	104	3	19196.16077	0.020
38400	32000000	52.08333333	52	1	38415.36615	0.040
57600	32000000	34.72222222	34	12	57553.95683	0.080
76800	32000000	26.04166667	26	1	76738.60911	0.080
115200	32000000	17.36111111	17	6	115107.9137	0.080

14.3 Memory Mapped IO Registers

Registers listed are for UART A, starting at base address B0002000h. UART 1 or UART B contains the same registers starting at base address B0002400h. Differences between the UARTs are noted in individual registers.

Table 40. Summary of UART Registers—0xB0002000

MEM Address	Default	Instance Name	Name
0xB0002000	0000_0000h	RBR_THR_DLL	Receive Buffer / Transmit Holding / Divisor Latch Low
0xB0002004	0000_0000h	IER_DLH	Interrupt Enable / Divisor Latch High
0xB0002008	0000_0001h	IIR_FCR	Interrupt Identification / FIFO Control
0xB000200C	0000_0000h	LCR	Line Control
0xB0002010	0000_0000h	MCR	MODEM Control
0xB0002014	0000_0060h	LSR	Line Status
0xB0002018	0000_0000h	MSR	MODEM Status
0xB000201C	0000_0000h	SCR	Scratchpad
0xB000207C	0000_0000h	USR	UART Status
0xB00020A4	0000_0000h	HTX	Halt Transmission
0xB00020A8	0000_0000h	DMASA	DMA Software Acknowledge
0xB00020AC	0000_0006h	TCR	Transceiver Control Register
0xB00020B0	0000_0000h	DE_EN	Driver Output Enable Register
0xB00020B4	0000_0000h	RE_EN	Receiver Output Enable Register
0xB00020B8	0000_0000h	DET	Driver Output Enable Timing Register
0xB00020BC	0000_0000h	TAT	TurnAround Timing Register
0xB00020C0	0000_0000h	DLF	Divisor Latch Fraction
0xB00020C4	0000_0000h	RAR	Receive Address Register
0xB00020C8	0000_0000h	TAR	Transmit Address Register
0xB00020CC	0000_0000h	LCR_EXT	Line Extended Control Register

14.3.1.1 Receive Buffer / Transmit Holding / Divisor Latch Low (RBR_THR_DLL)

Receive Buffer Register (RBR), reading this register when the DLAB bit (LCR[7]) is zero; Transmit Holding Register (THR), writing to this register when the DLAB is zero; Divisor Latch Low (DLL), when DLAB bit is one.

MEM Offset (B0002000)	0B0002000h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:9	RO	23'b0	**Reserved (RSV)**
8	RW	1'h0	**Receive Buffer / Transmit Holding (FIELD2)** Different UART registers are accessed depending on read/write transfer type. Transmit Holding register (MSB 9th bit). Access - Write Data byte received on the serial input port (sin) in UART mode for the MSB 9th bit. Receive Buffer register (MSB 9th bit). Access - Read Data byte received on the serial input port (sin) in UART mode, or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line Status Register (LSR) is set.

Bits	Access Type	Default	Description
7:0	RW	8'h00	**Receive Buffer / Transmit Holding / Divisor Latch Low (FIELD)** Different UART registers are accessed depending on read/write transfer type and Line control Register (LCR) DLAB bit value. RBR, Receive Buffer Register Access - Read AND DLAB (LCR[7]) =0 Data byte received on the serial input port (sin) in UART mode, or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line Status Register (LSR) is set THR, Transmit Holding Register. Access - Write AND DLAB (LCR[7]) =0 Data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. DLL, Lower part of the Divisor Latch. Access - Read/Write AND DLAB (LCR[7]) =1 Lower 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may be accessed only when the DLAB bit (LCR[7]) is set. Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of sclk should be allowed to pass before transmitting or receiving data.

14.3.1.2 Interrupt Enable / Divisor Latch High (IER_DLH)

Interrupt Enable Register (IER), when the DLAB bit is zero; Divisor Latch High (DLH), when the DLAB bit is one.

MEM Offset (B0002000)	0B0002004h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7:0	RW	8'h00	**Interrupt Enable / Divisor Latch High (FIELD)** Different UART registers are accessed depending on the Line control Register (LCR) DLAB bit value. IER, Interrupt Enable Register Access - Read/write AND DLAB (LCR[7]) =0 Interrupt Enable Register: Each of the bits used has a different function (0 = disabled 1 = enabled): - 0 ERBFI, Enable Received Data Available Interrupt - 1 ETBEI, Enable Transmit Holding Register Empty Interrupt - 2 ELSI, Enable Receiver Line Status Interrupt - 3 EDSSI, Enable Modem Status Interrupt - 4 RESERVED - 5 RESERVED - 6 RESERVED - 7 PTIME, Programmable THRE Interrupt Mode Enable DLH, Divisor Latch (High) Access - Read/write AND DLAB (LCR[7]) =1 This register makes up the upper 8-bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set. This register may be accessed only when the DLAB bit (LCR[7]) is set. Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of sclk should be allowed to pass before transmitting or receiving data.

14.3.1.3 Interrupt Identification / FIFO Control (IIR_FCR)

Interrupt Identification Register (IIR) if reading; FIFO Control Register (FCR) if writing.

MEM Offset (B0002000)	0B0002008h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7:0	RW	8'h01	**Interrupt Identification / FIFO Control (FIELD)** Different UART registers are accessed depending on read/write transfer type. IIR, Interrupt Identification Register Access - Read only Each of the bits used has a different function: 0-3 IID, Interrupt ID. This indicates the highest priority pending interrupt which can be one of the following types: 0000 = modem status. 0001 = no interrupt pending. 0010 = THR empty. 0100 = received data available. 0110 = receiver line status. 0111 = busy detect. NEVER INDICATED 1100 = character timeout. 4-5 RESERVED read as zero 6-7 FIFOSE, FIFOs Enabled. This is used to indicate whether the FIFO's are enabled or disabled: 00 = disabled 11 = enabled RESET VALUE FOR IIR = 0x01 FCR, FIFO Control Register Access - Write only Used to control the FIFOs. Different functions: 0 FIFOE, FIFO Enable. Enables/disables the transmit (XMIT) and receive (RCVR) FIFO's. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFO's will be reset. 1 RFIFOR, RCVR FIFO Reset Resets the control portion of the receive FIFO and treats the FIFO as empty. This will also de-assert the DMA RX request and single signals. NOTE that this bit is 'self-clearing' and it is not necessary to clear this bit. 2 XFIFOR, XMIT FIFO Reset Resets the control portion of the transmit FIFO and treats the FIFO as empty. This will also de-assert the DMA TX request and single signals. NOTE that this bit is 'self-clearing' and it is not necessary to clear this bit.

Bits	Access Type	Default	Description
			3 DMAM, DMA Mode 4-5 TET, TX Empty Trigger Used to select the empty threshold level at which the THRE Interrupts will be generated when the mode is active. It also determines when the dma_tx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO 1/4 full 11 = FIFO 1/2 full 6-7 RT, RCVR Trigger Used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt will be generated. In auto flow control mode it is used to determine when the rts_n signal will be de-asserted. It also determines when the dma_rx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO 1/4 full 10 = FIFO 1/2 full 11 = FIFO 2 less than full

14.3.1.4 Line Control (LCR)

Used to specify the format of the asynchronous data communication exchange.

MEM Offset (B0002000)	0B000200Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7	RW	1'h0	**Divisor Latch Access Bit (DLAB)** Used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.
6	RW	1'h0	**Break Control Bit (BC)** Used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared.

Bits	Access Type	Default	Description
5	RW	1'h0	**Reserved for future use (STICK_PAR)**
4	RW	1'h0	**Even Parity Select (EPS)** Used to select between even and odd parity, when parity is enabled (PEN set to one). If set to one, an even number of logic '1's is transmitted or checked. If set to zero, an odd number of logic '1's is transmitted or checked.
3	RW	1'h0	**Parity Enable (PEN)** Used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0 = parity disabled 1 = parity enabled
2	RW	1'h0	**Number of stop bits (STOP)** Used to select the number of stop bits per character that the peripheral will transmit and receive. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits is transmitted. Otherwise, two stop bits are transmitted. NOTE that regardless of the number of stop bits selected the receiver will only check the first stop bit. 0 = 1 stop bit 1 = 1.5 stop bits when DLS (LCR[1:0]) == 00 1 = 2 stop bits when DLS (LCR[1:0]) different from 00
1:0	RW	2'h0	**Data Length Select (DLS)** Data Length Select. If UART_16550_COMPATIBLE = NO, then writeable only when UART is not busy (USR[0] is 0); otherwise always writable, always readable. When DLS_E in LCR_EXT is set to 0, this is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected are as follows: 00 5 bits 01 6 bits 10 7 bits 11 8 bits

14.3.1.5 MODEM Control (MCR)

Controls the interface with the MODEM or data set (or a peripheral device emulating a MODEM).

MEM Offset (B0002000)	0B0002010h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:6	RO	26'b0	**Reserved (RSV)**
5	RW	1'h0	**Auto Flow Control Enable (AFCE)** When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, Auto Flow Control features are enabled. 0 = Auto Flow Control Mode disabled 1 = Auto Flow Control Mode enabled
4	RW	1'h0	**LoopBack Bit (LOOPBACK)** Used to put the UART into a diagnostic mode for test purposes. Data on the sout line is held high, while serial data output is looped back to the sin line, internally. In this mode all the interrupts are fully functional.
3	RW	1'h0	**User designated Output 2 (OUT2)** Used to directly control the user-designated Output2 (out2_n) output. The value written to this location is inverted and driven out on out2_n, that is: 0 = out2_n de-asserted (logic 1) 1 = out2_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out2_n output is held inactive high while the value of this location is internally looped back to an input.
2	RW	1'h0	**User designated Output 1 (OUT1)** Used to directly control the user-designated Output1 (out1_n) output. The value written to this location is inverted and driven out on out1_n, that is: 0 = out1_n de-asserted (logic 1) 1 = out1_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out1_n output is held inactive high while the value of this location is internally looped back to an input.

Bits	Access Type	Default	Description
1	RW	1'h0	**Request to Send (RTS)** Used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto RTS Flow Control is not enabled (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. In Auto Flow Control, MCR[5] set to one and FIFO's enable (FCR[0] set to one, the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal will be de-asserted when MCR[1] is set low. Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive high while the value of this location is internally looped back to an input.
0	RW	1'h0	**Data Terminal Ready (DTR)** Used to directly control the Data Terminal Ready (dtr_n) output. The value written to this location is inverted and driven out on dtr_n, that is: 0 = dtr_n de-asserted (logic 1) 1 = dtr_n asserted (logic 0) The Data Terminal Ready output is used to inform the modem or data set that the UART is ready to establish communications. Note that in Loopback mode (MCR[4] set to one), the dtr_n output is held inactive high while the value of this location is internally looped back to an input.

14.3.1.6 Line Status (LSR)

Provides status information concerning the data transfer.

MEM Offset (B0002000)	0B0002014h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0060h

Bits	Access Type	Default	Description
31:9	RO	23'b0	**Reserved (RSV)**

Bits	Access Type	Default	Description
8	RO/C	1'h0	**Address Received bit (ADDR_RCVD)** Address Received bit If 9Bit data mode (LCR_EXT[0]=1) is enabled, This bit is used to indicate the 9th bit of the receive data is set to 1. This bit can also be used to indicate whether the incoming character is address or data. 1 = Indicates the character is address. 0 = Indicates the character is data. In the FIFO mode, since the 9th bit is associated with a character received, it is revealed when the character with the 9th bit set to 1 is at the top of the FIFO. Reading the LSR clears the 9BIT. NOTE: User needs to ensure that interrupt gets cleared (reading LSR register) before the next address byte arrives. If there is a delay in clearing the interrupt, then Software will not be able to distinguish between multiple address related interrupt.
7	RO	1'h0	**Receiver FIFO Error bit (RFE)** This bit is only relevant when FIFO are enabled (FCR[0] set to one). This is used to indicate if there is at least one parity error, framing error, or break indication in the FIFO. That is: 0 = no error in RX FIFO 1 = error in RX FIFO This bit is cleared when the LSR is read and the character with the error is at the top of the receiver FIFO and there are no subsequent errors in the FIFO.
6	RO	1'h1	**Transmitter Empty bit (TEMT)** If FIFO are enabled (FCR[0] set to one), this bit is set whenever the Transmitter Shift Register and the FIFO are both empty. If FIFO are disabled, this bit is set whenever the Transmitter Holding Register and the Transmitter Shift Register are both empty.
5	RO	1'h1	**Transmit Holding Register Empty bit (THRE)** If THRE mode is disabled (IER[7] set to zero) and regardless of FIFO's being enabled or not, this bit indicates that the THR or TX FIFO is empty. This bit is set whenever data is transferred from the THR or TX FIFO to the transmitter shift register and no new data has been written to the THR or TX FIFO. This also causes a THRE Interrupt to occur, if the THRE Interrupt is enabled. If the THRE mode and FIFO are enabled (IER[7] and FCR[0] set to one), the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the FCR[5:4] threshold setting.
4	RO	1'h0	**Break Interrupt bit (BI)** Used to indicate the detection of a break sequence on the serial input data. If in UART mode it is set whenever the serial input, sin, is held in a logic '0' state for longer than the sum of start time + data bits + parity + stop bits. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.

Bits	Access Type	Default	Description
3	RO	1'h0	**Framing Error bit (FE)** Used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data. In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs the UART will try resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) will be set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no framing error, 1 = framing error Reading the LSR clears the FE bit.
2	RO	1'h0	**Parity Error bit (PE)** Used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) will be set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no parity error, 1 = parity error Reading the LSR clears the PE bit.
1	RO	1'h0	**Overrun error bit (OE)** Used to indicate the occurrence of an overrun error. This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost. 0 = no overrun error, 1 = overrun error Reading the LSR clears the OE bit.
0	RO	1'h0	**Data Ready bit (DR)** Used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0 = no data ready, 1 = data ready This bit is cleared when the RBR is read in the non-FIFO mode, or when the receiver FIFO is empty, in the FIFO mode.

14.3.1.7 MODEM Status (MSR)

Provides the current state of the control lines from the MODEM (or peripheral device).

MEM Offset (B0002000)	0B0002018h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7	RO	1'h0	**Data Carrier Detect (DCD)** Used to indicate the current state of the modem control line dcd_n. That is this bit is the complement dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set. 0 = dcd_n input is de-asserted (logic 1) 1 = dcd_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DCD is the same as MCR[3] (Out2).
6	RO	1'h0	**Ring Indicator (RI)** Used to indicate the current state of the modem control line ri_n. That is this bit is the complement ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0 = ri_n input is de-asserted (logic 1) 1 = ri_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), RI is the same as MCR[2] (Out1).
5	RO	1'h0	**Data Set Ready (DSR)** Used to indicate the current state of the modem control line dsr_n. That is this bit is the complement dsr_n. When the Data Set Ready input (dsr_n) is asserted it is an indication that the modem or data set is ready to establish communications with the UART. 0 = dsr_n input is de-asserted (logic 1) 1 = dsr_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DSR is the same as MCR[0] (DTR).
4	RO	1'h0	**Clear to Send (CTS)** Used to indicate the current state of the modem control line cts_n. That is, this bit is the complement cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with the UART. 0 = cts_n input is de-asserted (logic 1) 1 = cts_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), CTS is the same as MCR[1] (RTS).

Bits	Access Type	Default	Description
3	RO	1'h0	**Delta Data Carrier Detect (DDCD)** Used to indicate that the modem control line dcd_n has changed since the last time the MSR was read. That is: 0 = no change on dcd_n since last read of MSR 1 = change on dcd_n since last read of MSR Reading the MSR clears the DDCD bit. In Loopback Mode (MCR[4] set to one), DDCD reflects changes on MCR[3] (Out2). Note, if the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit will get set when the reset is removed if the dcd_n signal remains asserted.
2	RO	1'h0	**Trailing Edge of Ring Indicator (TERI)** Used to indicate that a change on the input ri_n (from an active low, to an inactive high state) has occurred since the last time the MSR was read. That is: 0 = no change on ri_n since last read of MSR 1 = change on ri_n since last read of MSR Reading the MSR clears the TERI bit. In Loopback Mode (MCR[4] set to one), TERI reflects when MCR[2] (Out1) has changed state from a high to a low.
1	RO	1'h0	**Delta Data Set Ready (DDSR)** Used to indicate that the modem control line dsr_n has changed since the last time the MSR was read. That is: 0 = no change on dsr_n since last read of MSR 1 = change on dsr_n since last read of MSR Reading the MSR clears the DDSR bit. In Loopback Mode (MCR[4] set to one), DDSR reflects changes on MCR[0] (DTR). Note, if the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDSR bit will get set when the reset is removed if the dsr_n signal remains asserted.
0	RO	1'h0	**Delta Clear to Send (DCTS)** Used to indicate that the modem control line cts_n has changed since the last time the MSR was read. That is: 0 = no change on cts_n since last read of MSR 1 = change on cts_n since last read of MSR Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] set to one), DCTS reflects changes on MCR[1] (RTS). Note, if the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit will get set when the reset is removed if the cts_n signal remains asserted.

14.3.1.8 Scratchpad (SCR)

Used by the programmer to hold data temporarily.

MEM Offset (B0002000)	0B000201Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**Reserved (RSV)**
7:0	RW	8'h0	**Scratchpad Register (SCR)** This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART.

14.3.1.9 UART Status (USR)

Provides internal status information.

MEM Offset (B0002000)	0B000207Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'h0	**Reserved (RSV1)**
4	RO	1'h0	**Receive FIFO Full (RFF)** Used to indicate that the receive FIFO is completely full. That is: 0 = Receive FIFO not full 1 = Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.
3	RO	1'h0	**Receive FIFO Not Empty (RFNE)** Used to indicate that the receive FIFO contains one or more entries. 0 = Receive FIFO is empty 1 = Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.
2	RO	1'h0	**Transmit FIFO Empty (TFE)** Used to indicate that the transmit FIFO is completely empty. 0 = Transmit FIFO is not empty 1 = Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty.
1	RO	1'h0	**Transmit FIFO Not Full (TFNF)** Used to indicate that the transmit FIFO in not full. 0 = Transmit FIFO is full 1 = Transmit FIFO is not full This bit is cleared when the TX FIFO is full.
0	RO	1'h0	**Reserved (RSV0)**

14.3.1.10 Halt Transmission (HTX)

Halt Transmission.

MEM Offset (B0002000)	0B00020A4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'h0	**Halt Transmission (HTX)** Used to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFO's are enabled. Note, if FIFO's are not enabled the setting of the halt TX register will have no effect on operation. 0 = Halt TX disabled 1 = Halt TX enabled

14.3.1.11 DMA Software Acknowledge (DMASA)

DMA software acknowledge if a transfer needs to be terminated due to an error condition.

MEM Offset (B0002000)	0B00020A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'h0	**DMA Software Acknowledge (DMASA)** Used to perform DMA software acknowledge if a transfer needs to be terminated due to an error condition. For example, if the DMA disables the channel, then the UART should clear its request. This will cause the TX request, TX single, RX request and RX single signals to de-assert. Note that this bit is 'self-clearing' and it is not necessary to clear this bit.

14.3.1.12 Transceiver Control Register (TCR)

Transceiver Control Register.

MEM Offset (B0002000)	0B00020ACh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0006h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**
4:3	RW	2'h0	**Transfer Mode (XFER_MODE)** Transfer Mode 0 : In this mode, transmit and receive can happen simultaneously. The user can enable DE_EN, RE_EN at any point of time. Turn around timing as programmed in the TAT register is not applicable in this mode. 1 : In this mode, DE and RE are mutually exclusive. Either DE or RE only one of them is expected to be enabled through programming. Hardware will consider the Turn Around timings which are programmed in the TAT register while switching from RE to DE or DE to RE. For transmission Hardware will wait if it is in middle of receiving any transfer, before it starts transmitting. 2 : In this mode, DE and RE are mutually exclusive. Once DE_EN/RE_EN is programed - by default re will be enabled and UART Controller controller will be ready to receive. If the user programs the TX FIFO with the data then the UART Controller, after ensuring no receive is in progress, disables the RE signal and enables the DE signal. Once the TX FIFO becomes empty, RE signal gets enabled and DE signal will be disabled. In this mode of operation hardware will consider the Turn Around timings which are programmed in the TAT register while switching from RE to DE or DE to RE. In this mode, DE and RE signals are strictly complementary to each other.
2	RW	1'h1	**Driver Enable Polarity (DE_POL)** Driver Enable Polarity. 0: DE signal is active low 1: DE signal is active high
1	RW	1'h1	**Receiver Enable Polarity (RE_POL)** Receiver Enable Polarity. 1: RE signal is active high 0: RE signal is active low
0	RW	1'h0	**RS485 Transfer Enable (RS485_EN)** RS485 Transfer Enable. 0 : In this mode, the transfers are still in the RS232 mode. All other fields in this register are reserved and registers DE_EN, RE_EN, DET and TAT are reserved. 1 : In this mode, the transfers will happen in RS485 mode. All other fields of this register are applicable.

14.3.1.13 Driver Output Enable Register (DE_EN)

Driver Output Enable Register.

MEM Offset (B0002000)	0B00020B0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'h0	**Driver Output Enable (DE_Enable)** The DE Enable register bit is used to control assertion and de-assertion of de signal. 0: De-assert de signal 1: Assert de signal

14.3.1.14 Receiver Output Enable Register (RE_EN)

Receiver Output Enable Register.

MEM Offset (B0002000)	0B00020B4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'h0	**Receiver Output Enable (RE_Enable)** The RE Enable register bit is used to control assertion and de-assertion of re signal. 0: De-assert RE signal 1: Assert RE signal

14.3.1.15 Driver Output Enable Timing Register (DET)

Used to holds the DE assertion and de-assertion timings of the signal.

MEM Offset (B0002000)	0B00020B8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:24	RO	8'b0	**Reserved (RSV1)**
23:16	RW	8'h0	**DE de-assertion time (DE_deassertion_time)** DE signal de-assertion time. This field controls the amount of time (in terms of number of serial clock periods) between the end of stop bit on the sout to the falling edge of Driver output enable signal.
15:8	RO	8'b0	**Reserved (RSV0)**
7:0	RW	8'h0	**DE assertion time (DE_assertion_time)** DE signal assertion time. This field controls the amount of time (in terms of number of serial clock periods) between the assertion of rising edge of Driver output enable signal to serial transmit enable. Any data in transmit buffer, will start on serial output (sout) after the transmit enable.

14.3.1.16 TurnAround Timing Register (TAT)

TurnAround Timing Register.

MEM Offset (B0002000)	0B00020BCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RW	16'b0	**DE to RE (RE_to_DE)** Driver Enable to Receiver Enable TurnAround time. Turnaround time (in terms of serial clock) for DE de-assertion to RE assertion.
15:0	RW	16'h0	**DE to RE (DE_to_RE)** Driver Enable to Receiver Enable TurnAround time. Turnaround time (in terms of serial clock) for DE de-assertion to RE assertion.

14.3.1.17 Divisor Latch Fraction (DLF)

Divisor Latch Fraction.

MEM Offset (B0002000)	0B00020C0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved (RSV)**
3:0	RW	4'h0	**Divisor Latch Fraction (DLF)** Register Divisor Latch Fraction (DLF) is used to store the Fractional part of BAUD divisor.

14.3.1.18 Receive Address Register (RAR)

Receive Address Register.

MEM Offset (B0002000)	0B00020C4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**RSRV (RSRV)**
7:0	RW	8'h0	**RAR (RAR)** This is an address matching register during receive mode. If the 9-th bit is set in the incoming character then the remaining 8-bits will be checked against this register value. If the match happens then sub-sequent characters with 9-th bit set to 0 will be treated as data byte until the next address byte is received. NOTE: This register is applicable only when ADDR_MATCH (LCR[9]) and 'DLS_E' (LCR[8]) bits are set to 1.

14.3.1.19 Transmit Address Register (TAR)

Transmit Address Register.

MEM Offset (B0002000)	0B00020C8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'b0	**RSRV (RSRV)**

Bits	Access Type	Default	Description
7:0	RW	8'h0	**TAR (TAR)** This is an address matching register during transmit mode. If DLS_E (LCR[8]) bit is enabled, then the UART Controller will send the 9-bit character with 9-th bit set to 1 and remaining 8-bit address will be sent from this register provided 'SEND_ADDR' (LCR[10]) bit is set to 1. NOTE: 1. This register is used only to send the address. The normal data should be sent by programming THR register. 2. Once the address is started to send on the UART Controller serial lane, then SEND_ADDR bit will be auto-cleared by the hardware.

14.3.1.20 Line Extended Control Register (LCR_EXT)

Line Extended Control Register.

MEM Offset (B0002000)	0B00020CCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved (RSV)**
3	RW	1'h0	**Transmit mode control bit (TRANSMIT_MODE)** Transmit mode control bit. This bit is used to control the type of transmit mode during 9-bit data transfers. 1 : In this mode of operation, Transmit Holding Register (THR) and Shadow Transmit Holding Register (STHR) are 9-bit wide. The user needs to ensure that the THR/STHR register is written correctly for address/data. Address: 9th bit is set to 1, Data : 9th bit is set to 0. NOTE: Transmit address register (TAR) is not applicable in this mode of operation. 0: In this mode of operation, Transmit Holding Register (THR) and Shadow Transmit Holding register (STHR) are 8-bit wide. The user needs to program the address into Transmit Address Register (TAR) and data into the THR/STHR register. SEND_ADDR (LCR_EXT[10]) bit is used as a control knob to indicate the UART Controller on when to send the address.

Bits	Access Type	Default	Description
2	RW/AC	1'h0	**Send address control bit (SEND_ADDRESS)** Send address control bit. This bit is used as a control knob for the user to determine when to send the address during transmit mode. 1 = 9-bit character will be transmitted with 9-th bit set to 1 and the remaining 8-bits will match to what is being programmed in Transmit Address Register. 0 = 9-bit character will be transmitted with 9-th bit set to 0 and the remaining 8-bits will be taken from the TXFIFO which is programmed through 8-bit wide THR/STHR register. NOTE: 1. This bit is auto-cleared by the hardware, after sending out the address character. User is not expected to program this bit to 0. 2. This field is applicable only when DLS_E bit is set to 1 and TRANSMIT_MODE is set to 0.
1	RW	1'h0	**Address Match Mode (ADDR_MATCH)** Address Match Mode.This bit is used to enable the address match feature during receive. 1 = Address match mode; UART Controller will wait until the incoming character with 9-th bit set to 1. And further checks to see if the address matches with what is programmed in Receive Address Match Register. If match is found, then sub-sequent characters will be treated as valid data and UART Controller starts receiving data. 0 = Normal mode; UART Controller will start to receive the data and 9-bit character will be formed and written into the receive RXFIFO. User is responsible to read the data and differentiate b/n address and data. NOTE: This field is applicable only when DLS_E is set to 1.
0	RW	1'h0	**Extension for DLS (DLS_E)** Extension for DLS. This bit is used to enable 9-bit data for transmit and receive transfers. 1 = 9 bits per character 0 = Number of data bits selected by DLS

§

15 SPI

The SoC implements two instances of a SPI controller. One controller supports Master operation and another controller supports Slave operation.

15.1 Signal Descriptions

Please see Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, "Electrical Characteristics"
- **Description:** A brief explanation of the signal's function

Table 41. SPI Master 0 Signals

Signal Name	Direction/ Type	Description
SPI_M_SCLK	Logic output	Master SPI Clock
SPI_M_TXD	Logic output	Master SPI Transmit data
SPI_M_SS[3:0]	Logic output	Master SPI Slave Selects
SPI_M_RXD	Logic input	Master SPI Receive data

Table 42. SPI Slave 0 Signals

Signal Name	Direction/ Type	Description
SPI_S_SCLK	Logic input	Slave SPI Clock
SPI_S_SDIN	Logic input	Slave SPI Receive data
SPI_S_SCS	Logic input	Slave SPI Slave Chip Select
SPI_S_SDOUT	Logic output	Slave SPI Transmit data

15.2 Features

The following is a list of the SPI Master features:

- One SPI Master Interface
- Control of up to 4 Slave Selects
- Frame Formats:
  - Motorola SPI*
- Transfer Modes:
  - Transmit & Receive
  - Transmit Only
  - Receive Only
  - EEPROM Read
- Serial Clock Frequencies up to 16 MHz
- 4 bit to 32 bit Frame Size
- Configurable Clock Polarity and Clock Phase
- Hardware Handshake Interface to support DMA capability
- Interrupt Control
- FIFO mode support with 8B deep TX and RX FIFO's

The following is a list of the SPI Slave features:

- One SPI Slave Interface
- Frame Formats:
  - Motorola SPI*

  - Transfer Modes:
  - Transmit & Receive
  - Transmit Only
  - Receive Only
  - EEPROM Read
- Serial Clock Frequencies up to 3.2 MHz
- 4 bit to 32 bit Frame Size
- Configurable Clock Polarity and Clock Phase
- Hardware Handshake Interface to support DMA capability
- Interrupt Control
- FIFO mode support with 8B deep TX and RX FIFO's

15.3 Memory Mapped IO Registers

Registers listed are for SPI Master 0, starting at base address B0001000h.
SPI Slave 0 contains the same registers starting at base address B0001800h. Differences between the SPIs are noted in individual registers.

Table 43. Summary of SPI Registers—0xB0001000 & 0xB0001800

MEM Address	Default	Instance Name	Name
0x00	0007_0000h	CTRLR0	Control Register 0
0x04	0000_0000h	CTRLR1	Control Register 1
0x08	0000_0000h	SSIENR	SSI Enable Register
0x0C	0000_0000h	MWCR	Microwire Control Register
0x10	0000_0000h	SER	Slave Enable Register
0x14	0000_0000h	BAUDR	Baud Rate Select
0x18	0000_0000h	TXFTLR	Transmit FIFO Threshold Level
0x1C	0000_0000h	RXFTLR	Receive FIFO Threshold Level
0x20	0000_0000h	TXFLR	Transmit FIFO Level Register
0x24	0000_0000h	RXFLR	Receive FIFO Level Register
0x28	0000_0006h	SR	Status Register
0x2C	0000_003Fh	IMR	Interrupt Mask Register
0x30	0000_0000h	ISR	Interrupt Status Register
0x34	0000_0000h	RISR	Raw Interrupt Status Register
0x38	0000_0000h	TXOICR	Transmit FIFO Overflow Interrupt Clear Register
0x3C	0000_0000h	RXOICR	Receive FIFO Overflow Interrupt Clear Register
0x40	0000_0000h	RXUICR	Receive FIFO Underflow Interrupt Clear Register
0x44	0000_0000h	MSTICR	Multi-Master Interrupt Clear Register
0x48	0000_0000h	ICR	Interrupt Clear Register
0x4C	0000_0000h	DMACR	DMA Control Register
0x50	0000_0000h	DMATDLR	DMA Transmit Data Level
0x54	0000_0000h	DMARDLR	DMA Receive Data Level
0x58	0000_0000h	IDR	Identification Register
0x5C	3332_332Ah	SSI_COMP_VERSION	coreKit Version ID register
0x60	0000_0000h	DR0	Data Register
0x64	0000_0000h	DR1	Data Register
0x68	0000_0000h	DR2	Data Register
0x6C	0000_0000h	DR3	Data Register
0x70	0000_0000h	DR4	Data Register
0x74	0000_0000h	DR5	Data Register

MEM Address	Default	Instance Name	Name
0x78	0000_0000h	DR6	Data Register
0x7C	0000_0000h	DR7	Data Register
0x80	0000_0000h	DR8	Data Register
0x84	0000_0000h	DR9	Data Register
0x88	0000_0000h	DR10	Data Register
0x8C	0000_0000h	DR11	Data Register
0x90	0000_0000h	DR12	Data Register
0x94	0000_0000h	DR13	Data Register
0x98	0000_0000h	DR14	Data Register
0x9C	0000_0000h	DR15	Data Register
0xA0	0000_0000h	DR16	Data Register
0xA4	0000_0000h	DR17	Data Register
0xA8	0000_0000h	DR18	Data Register
0xAC	0000_0000h	DR19	Data Register
0xB0	0000_0000h	DR20	Data Register
0xB4	0000_0000h	DR21	Data Register
0xB8	0000_0000h	DR22	Data Register
0xBC	0000_0000h	DR23	Data Register
0xC0	0000_0000h	DR24	Data Register
0xC4	0000_0000h	DR25	Data Register
0xC8	0000_0000h	DR26	Data Register
0xCC	0000_0000h	DR27	Data Register
0xD0	0000_0000h	DR28	Data Register
0xD4	0000_0000h	DR29	Data Register
0xD8	0000_0000h	DR30	Data Register
0xDC	0000_0000h	DR31	Data Register
0xE0	0000_0000h	DR32	Data Register
0xE4	0000_0000h	DR33	Data Register
0xE8	0000_0000h	DR34	Data Register
0xEC	0000_0000h	DR35	Data Register
0xF0	0000_0000h	RX_SAMPLE_DLY	RX Sample Delay Register

15.3.1.1 Control Register 0 (CTRLR0)

This register controls the serial data transfer. It is impossible to write to this register when the SPI Controller is enabled. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	00h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0007_0000h

Bits	Access Type	Default	Description
31:21	RO	11'b0	**Reserved 2 (RSVD2)** Reserved
20:16	RW/L	5'h07	**Data Frame Size in 32-bit mode (DFS_32)** Used to select the data frame length in 32 bit mode. These bits are only valid when SSI_MAX_XFER_SIZE is configured to 32. When the data frame size is programmed to be less than 32-bits, the receive data is automatically right-justified by the receive logic, with the upper bits of the receive FIFO zero-padded. Transmit data must be right-justified by the user before writing into the transmit FIFO. The transmit logic will ignore the upper unused bits when transmitting the data.
15:12	RW/L	4'h0	**Control Frame Size (CFS)** Control Frame Size. Selects the length of the control word for the Microwire* frame format.
11	RW/L	1'h0	**Shift Register Loop (SRL)** Used for testing purposes only. When internally active, connects the transmit shift register output to the receive shift register input. Can be used in both serial-slave and serial-master modes. 0 : Normal Mode Operation 1 : Test Mode Operation When the SPI Controller is configured as a slave in loopback mode, the ss_in_n and ssi_clk signals must be provided by an external source. In this mode, the slave cannot generate these signals because there is nothing to which to loop back.
10	RW/L	1'h0	**Slave Output Enable (SLV_OE)** Relevant only when the SPI Controller is configured as a serial-slave device. When configured as a serial master, this bit field has no functionality. This bit enables or disables the setting of the ssi_oe_n output from the SPI Controller serial slave. When SLV_OE = 1, the ssi_oe_n output can never be active. When the ssi_oe_n output controls the tri-state buffer on the txd output from the slave, a high impedance state is always present on the slave txd output when SLV_OE = 1. This is useful when the master transmits in broadcast mode (master transmits data to all slave devices). Only one slave may respond with data on the master rxd line. This bit is enabled after reset and must be disabled by software (when broadcast mode is used), if you do not want this device to respond with data. 0 - Slave txd is enabled 1 - Slave txd is disabled

Bits	Access Type	Default	Description
9:8	RW/L	2'h0	**Transfer Mode (TMOD)** Transfer Mode. Selects the mode of transfer for serial communication. This field does not affect the transfer duplicity. Only indicates whether the receive or transmit data is valid. In transmit-only mode, data received from the external device is not valid and is not stored in the receive FIFO memory; it is overwritten on the next transfer. In receive-only mode, transmitted data is not valid. After the first write to the transmit FIFO, the same word is retransmitted for the duration of the transfer. In transmit-and-receive mode, both transmit and receive data is valid. The transfer continues until the transmit FIFO is empty. Data received from the external device are stored into the receive FIFO memory, where it can be accessed by the host processor. In eeprom-read mode, receive data is not valid while control data is being transmitted. When all control data is sent to the EEPROM, receive data becomes valid and transmit data becomes invalid. All data in the transmit FIFO is considered control data in this mode. This transfer mode is only valid when the SPI Controller is configured as a master device. 00 - Transmit & Receive 01 - Transmit Only 10 - Receive Only 11 - EEPROM Read
7	RW/L	1'h0	**Serial Clock Polarity (SCPOL)** Valid when the frame format (FRF) is set to Motorola SPI*. Used to select the polarity of the inactive serial clock, which is held inactive when the SPI Controller master is not actively transferring data on the serial bus. 0 : Inactive state of serial clock is low 1 : Inactive state of serial clock is high Dependencies: When SSI_HC_FRF=1, SCPOL bit is a read-only bit with its value set by SSI_DFLT_SCPOL.
6	RW/L	1'h0	**Serial Clock Phase (SCPH)** Valid when the frame format (FRF) is set to Motorola SPI*. The serial clock phase selects the relationship of the serial clock with the slave select signal. When SCPH = 0, data is captured on the first edge of the serial clock. When SCPH = 1, the serial clock starts toggling one cycle after the slave select line is activated, and data is captured on the second edge of the serial clock. 0: Serial clock toggles in middle of first data bit 1: Serial clock toggles at start of first data bit Dependencies: When SSI_HC_FRF=1, SCPH bit is a read-only bit, with its value set by SSI_DFLT_SCPH.
5:4	RW	2'h0	**Frame Format (FRF)** Selects which serial protocol transfers the data. b00 – Motorola SPI* b01 - Texas Instruments SSP* b10 - National Semiconductors Microwire* b11 – Reserved
3:0	RO	4'b0	**Reserved 1 (RSVD1)** Reserved

15.3.1.2 Control Register 1 (CTRLR1)

This register exists only when the SPI Controller is configured as a master device. When the SPI Controller is configured as a serial slave, writing to this location has no effect; reading from this location returns 0. Control register 1 controls the end of serial transfers when in receive-only mode. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	04h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved
15:0	RW/L	16'h0	**Number of Data Frames (NDF)** When TMOD = 10 or TMOD = 11, this register field sets the number of data frames to be continuously received by the SPI Controller. The SPI Controller continues to receive serial data until the number of data frames received is equal to this register value plus 1, which enables you to receive up to 64 KB of data in a continuous transfer. When the SPI Controller is configured as a serial slave, the transfer continues for as long as the slave is selected.

15.3.1.3 SSI Enable Register (SSIENR)

MEM Offset (B0001000)	08h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved 1 (RSVD1)** Reserved
0	RW	1'h0	**SSI Enable (SSIENR)** Enables and disables all SPI Controller operations. When disabled, all serial transfers are halted immediately. Transmit and receive FIFO buffers are cleared when the device is disabled. It is impossible to program some of the SPI Controller control registers when enabled. When disabled, the ssi_sleep output is set (after delay) to inform the system that it is safe to remove the ssi_clk, thus saving power consumption in the system.

15.3.1.4 Microwire Control Register (MWCR)

This register controls the direction of the data word for the half-duplex Microwire serial protocol. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved 1 (RSVD1)** Reserved
2	RW/L	1'h0	**Microwire Handshaking (MHS)** Relevant only when the SPI Controller is configured as a serial-master device. When configured as a serial slave, this bit field has no functionality. Used to enable and disable the 'busy/ready' handshaking interface for the Microwire protocol. When enabled, the SPI Controller checks for a ready status from the target slave, after the transfer of the last data/control bit, before clearing the BUSY status in the SR register. 0: handshaking interface is disabled 1: handshaking interface is enabled
1	RW/L	1'h0	**Microwire Control Register (MDD)** Defines the direction of the data word when the Microwire serial protocol is used. When this bit is set to 0, the data word is received by the SPI Controller from the external serial device. When this bit is set to 1, the data word is transmitted from the SPI Controller to the external serial device.
0	RW/L	1'h0	**Microwire Transfer Mode (MWMOD)** Defines whether the Microwire transfer is sequential or non-sequential. When sequential mode is used, only one control word is needed to transmit or receive a block of data words. When non-sequential mode is used, there must be a control word for each data word that is transmitted or received. 0 : non-sequential transfer 1 : sequential transfer

15.3.1.5 Slave Enable Register (SER)

This register is valid only when the SPI Controller is configured as a master device. When the SPI Controller is configured as a serial slave, writing to this location has no effect; reading from this location returns 0. The register enables the individual slave select output lines from the SPI Controller master. Up to 4 slave-select output signals are available on the SPI Controller master. You cannot write to this register when SPI Controller is busy.

MEM Offset (B0001000) 10h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved 1 (RSVD1)** Reserved
3:0	RW/L	4'h0	**Slave Select Enable Flag (SER)** Each bit in this register corresponds to a slave select line (ss_x_n]) from the SPI Controller master. When a bit in this register is set (1), the corresponding slave select line from the master is activated when a serial transfer begins. It should be noted that setting or clearing bits in this register have no effect on the corresponding slave select outputs until a transfer is started. Before beginning a transfer, you should enable the bit in this register that corresponds to the slave device with which the master wants to communicate. When not operating in broadcast mode, only one bit in this field should be set. 1: Selected 0: Not Selected

15.3.1.6 Baud Rate Select (BAUDR)

This register is valid only when the SPI Controller is configured as a master device. When the SPI Controller is configured as a serial slave, writing to this location has no effect; reading from this location returns 0. The register derives the frequency of the serial clock that regulates the data transfer. The 16-bit field in this register defines the ssi_clk divider value. It is impossible to write to this register when the SPI Controller is enabled. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000) 14h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW/L	16'h0	**SSI Clock Divider (SCKDV)**
			The LSB for this field is always set to 0 and is unaffected by a write operation, which ensures an even value is held in this register. If the value is 0, the serial output clock (sclk_out) is disabled. The frequency of the sclk_out is derived from the following equation:
			Fsclk_out = Fssi_clk/SCKDV
			where SCKDV is any even value between 2 and 65534. For example:
			for Fssi_clk = 3.6864MHz and SCKDV =2
			Fsclk_out = 3.6864/2 = 1.8432MHz

15.3.1.7 Transmit FIFO Threshold Level (TXFTLR)

This register controls the threshold value for the transmit FIFO memory. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	18h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved 1 (RSVD1)** Reserved
2:0	RW	3'b0	**Transmit FIFO Threshold (TXFTLR)** Controls the level of entries (or below) at which the transmit FIFO controller triggers an interrupt. If you attempt to set register field to a value greater than or equal to the depth of the FIFO, this field is not written and retains its current value. When the number of transmit FIFO entries is less than or equal to this value, the transmit FIFO empty interrupt is triggered.

15.3.1.8 Receive FIFO Threshold Level (RXFTLR)

This register controls the threshold value for the receive FIFO memory. The SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	1Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved 1 (RSVD1)** Reserved
2:0	RW	3'h0	**Receive FIFO Threshold (RFT)** Controls the level of entries (or above) at which the receive FIFO controller triggers an interrupt. If you attempt to set this value greater than the depth of the FIFO, this field is not written and retains its current value. When the number of receive FIFO entries is greater than or equal to this value + 1, the receive FIFO full interrupt is triggered.

15.3.1.9 Transmit FIFO Level Register (TXFLR)

MEM Offset (B0001000)	20h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved 1 (RSVD1)** Reserved
3:0	RO	4'h0	**Transmit FIFO Level (TXTFL)** Contains the number of valid data entries in the transmit FIFO.

15.3.1.10 Receive FIFO Level Register (RXFLR)

MEM Offset (B0001000)	24h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'b0	**Reserved 1 (RSVD1)** Reserved
3:0	RO	4'h0	**Receive FIFO Level (RXFLR)** Contains the number of valid data entries in the receive FIFO.

15.3.1.11 Status Register (SR)

This is a read-only register used to indicate the current transfer status, FIFO status, and any transmission/reception errors that may have occurred. The status register may be read at any time. None of the bits in this register request an interrupt.

MEM Offset (B0001000)	28h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0006h

Bits	Access Type	Default	Description
31:7	RO	25'h0	**Reserved 1 (RSVD1)** Reserved
6	RO	1'h0	**RSVD (RSVD)** Reserved
5	RO	1'h0	**Transmission Error (TXE)** Set if the transmit FIFO is empty when a transfer is started. This bit can be set only when the SPI Controller is configured as a slave device. Data from the previous transmission is resent on the txd line. This bit is cleared when read. 0 : No error 1 : Transmission error
4	RO	1'h0	**Receive FIFO Full (RFF)** When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0 : Receive FIFO is not full 1 : Receive FIFO is full
3	RO	1'h0	**Receive FIFO Not Empty (RFNE)** Set when the receive FIFO contains one or more entries and is cleared when the receive FIFO is empty. This bit can be polled by software to completely empty the receive FIFO. 0 : Receive FIFO is empty 1 : Receive FIFO is not empty
2	RO	1'h1	**Transmit FIFO Empty (TFE)** When the transmit FIFO is completely empty, this bit is set. When the transmit FIFO contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0 : Transmit FIFO is not empty 1 : Transmit FIFO is empty
1	RO	1'h1	**Transmit FIFO Not Full (TFNF)** Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0 : Transmit FIFO is full 1 : Transmit FIFO is not full
0	RO	1'h0	**SSI Busy Flag (BUSY)** When set, indicates that a serial transfer is in progress; when cleared indicates that the SPI Controller is idle or disabled. 0 : SPI Controller is idle or disabled 1 : SPI Controller is actively transferring data

15.3.1.12 Interrupt Mask Register (IMR)

This read/write register masks or enables all interrupts generated by the SPI Controller.

MEM Offset (B0001000)	2Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_003Fh

Bits	Access Type	Default	Description
31:6	RO	26'b0	**Reserved 1 (RSVD1)** Reserved
5	RO	1'h1	**RSVD (RSVD)** Reserved
4	RW	1'h1	**Receive FIFO Full Interrupt Mask (RXFIM)** 0 : ssi_rxf_intr interrupt is masked 1 : ssi_rxf_intr interrupt is not masked
3	RW	1'h1	**Receive FIFO Overflow Interrupt Mask (RXOIM)** 0 : ssi_rxo_intr interrupt is masked 1 : ssi_rxo_intr interrupt is not masked
2	RW	1'h1	**Receive FIFO Underflow Interrupt Mask (RXUIM)** 0 : ssi_rxu_intr interrupt is masked 1 : ssi_rxu_intr interrupt is not masked
1	RW	1'h1	**Transmit FIFO Overflow Interrupt Mask (TXOIM)** 0 : ssi_txo_intr interrupt is masked 1 : ssi_txo_intr interrupt is not masked
0	RW	1'h1	**Transmit FIFO Empty Interrupt Mask (TXEIM)** 0 : ssi_txe_intr interrupt is masked 1 : ssi_txe_intr interrupt is not masked

15.3.1.13 Interrupt Status Register (ISR)

This register reports the status of the SPI Controller interrupts after they have been masked.

MEM Offset (B0001000)	30h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:6	RO	26'b0	**Reserved 1 (RSVD1)** Reserved
5	RO	1'h0	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
4	RO	1'h0	**Receive FIFO Full Interrupt Status (RXFIS)** 0 : ssi_rxf_intr interrupt is not active after masking 1 : ssi_rxf_intr interrupt is active after masking
3	RO	1'h0	**Receive FIFO Overflow Interrupt Status (RXOIS)** 0 : ssi_rxo_intr interrupt is active after masking 1 : ssi_rxo_intr interrupt is not active after masking
2	RO	1'h0	**Receive FIFO Underflow Interrupt Status (RXUIS)** 0 : ssi_rxu_intr interrupt is not active after masking 1 : ssi_rxu_intr interrupt is active after masking
1	RO	1'h0	**Transmit FIFO Overflow Interrupt Status (TXOIS)** 0 : ssi_txo_intr interrupt is not active after masking 1 : ssi_txo_intr interrupt is active after masking
0	RO	1'h0	**Transmit FIFO Empty Interrupt Status (TXEIS)** 0 : ssi_txe_intr interrupt is not active after masking 1 : ssi_txe_intr interrupt is active after masking

15.3.1.14 Raw Interrupt Status Register (RISR)

This register reports the status of the SPI Controller interrupts prior to masking

MEM Offset (B0001000)	34h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:6	RO	26'b0	**Reserved 1 (RSVD1)** Reserved
5	RO	1'h0	**RSVD (RSVD)** Reserved
4	RO	1'h0	**Receive FIFO Full Raw Interrupt Status (RXFIR)** 0 : ssi_rxf_intr interrupt is not active prior masking 1 : ssi_rxf_intr interrupt is active prior masking
3	RO	1'h0	**Receive FIFO Overflow Raw Interrupt Status (RXOIR)** 0 : ssi_rxo_intr interrupt is active prior masking 1 : ssi_rxo_intr interrupt is not active prior masking
2	RO	1'h0	**Receive FIFO Underflow Raw Interrupt Status (RXUIR)** 0 : ssi_rxu_intr interrupt is not active prior masking 1 : ssi_rxu_intr interrupt is active prior masking

Bits	Access Type	Default	Description
1	RO	1'h0	**Transmit FIFO Overflow Raw Interrupt Status (TXOIR)** 0 : ssi_txo_intr interrupt is not active prior masking 1 : ssi_txo_intr interrupt is active prior masking
0	RO	1'h0	**Transmit FIFO Empty Raw Interrupt Status (TXEIR)** 0 : ssi_txe_intr interrupt is not active prior masking 1 : ssi_txe_intr interrupt is active prior masking

15.3.1.15 Transmit FIFO Overflow Interrupt Clear Register (TXOICR)

MEM Offset (B0001000)	38h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved 1 (RSVD1)** Reserved
0	RO/C	1'h0	**Clear Transmit FIFO Overflow Interrupt (TXOICR)** This register reflects the status of the interrupt. A read from this register clears the ssi_txo_intr interrupt; writing has no effect.

15.3.1.16 Receive FIFO Overflow Interrupt Clear Register (RXOICR)

MEM Offset (B0001000)	3Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO/C	31'b0	**Reserved 1 (RSVD1)** Reserved
0	RO/C	1'h0	**Clear Receive FIFO Overflow Interrupt (RXOICR)** This register reflects the status of the interrupt. A read from this register clears the ssi_rxo_intr interrupt; writing has no effect.

15.3.1.17 Receive FIFO Underflow Interrupt Clear Register (RXUICR)

MEM Offset (B0001000)	40h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved 1 (RSVD1)** Reserved
0	RO/C	1'h0	**Clear Receive FIFO Underflow Interrupt (RXUICR)** This register reflects the status of the interrupt. A read from this register clears the ssi_rxu_intr interrupt; writing has no effect.

15.3.1.18 Multi-Master Interrupt Clear Register (MSTICR)

MEM Offset (B0001000)	44h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved 1 (RSVD1)** Reserved
0	RO	1'h0	**RSVD (RSVD)** Reserved

15.3.1.19 Interrupt Clear Register (ICR)

MEM Offset (B0001000)	48h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description
0	RO/C	1'h0	**Interrupt Clear Register (ICR)** This register is set if any of the interrupts below are active. A read clears the ssi_txo_intr, ssi_rxu_intr, ssi_rxo_intr, and the ssi_mst_intr interrupts. Writing to this register has no effect.

15.3.1.20 DMA Control Register (DMACR)

The register is used to enable the DMA Controller interface operation.

MEM Offset (B0001000)	4Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'h0	**Reserved 1 (RSVD1)** Reserved
1	RW	1'h0	**Transmit DMA Enable (TDMAE)** This bit enables/disables the transmit FIFO DMA channel. 0 = Transmit DMA disabled 1 = Transmit DMA enabled
0	RW	1'h0	**Receive DMA Enable (RDMAE)** This bit enables/disables the receive FIFO DMA channel. 0 = Receive DMA disabled 1 = Receive DMA enabled

15.3.1.21 DMA Transmit Data Level (DMATDLR)

MEM Offset (B0001000)	50h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'h0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description
2:0	RW	3'h0	**DMA Transmit Data Level (DMATDL)** Transmit Data Level. This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the transmit FIFO is equal to or below this field value, and TDMAE = 1.

15.3.1.22 DMA Receive Data Level (DMARDLR)

MEM Offset (B0001000)	54h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'b0	**Reserved 1 (RSVD1)** Reserved
2:0	RW	3'h0	**Receive Data Level (DMARDL)** This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or above this field value + 1, and RDMAE=1.

15.3.1.23 Identification Register (IDR)

This read-only register is available for use to store a peripheral identification code.

MEM Offset (B0001000)	58h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO	32'h0	**Identification Code (IDCODE)** This register contains the peripherals identification code, which is written into the register at configuration time using coreConsultant.

15.3.1.24 coreKit Version ID register (SSI_COMP_VERSION)

This read-only register stores the specific SPI Controller component version.

MEM Offset (B0001000)	5Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	3332_332Ah

Bits	Access Type	Default	Description
31:0	RO	32'h3332332a	**SSI Component Version (SSI_COMP_VERSION)**

15.3.1.25 Data Register (DR0)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	60h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.26 Data Register (DR1)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	64h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.27 Data Register (DR2)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	68h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.28 Data Register (DR3)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	6Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.29 Data Register (DR4)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	70h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.30 Data Register (DR5)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	74h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.31 Data Register (DR6)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	78h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.32 Data Register (DR7)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	7Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.33 Data Register (DR8)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	80h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.34 Data Register (DR9)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000) 84h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.35 Data Register (DR10)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000) 88h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.36 Data Register (DR11)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	8Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.37 Data Register (DR12)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	90h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.38 Data Register (DR13)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	94h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.39 Data Register (DR14)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	98h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.40 Data Register (DR15)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	9Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.41 Data Register (DR16)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	A0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.42 Data Register (DR17)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	A4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.43 Data Register (DR18)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.44 Data Register (DR19)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	ACh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.45 Data Register (DR20)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	B0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.46 Data Register (DR21)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	B4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.47 Data Register (DR22)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	B8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.48 Data Register (DR23)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	BCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.49 Data Register (DR24)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	C0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.50 Data Register (DR25)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	C4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.51 Data Register (DR26)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	C8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.52 Data Register (DR27)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	CCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.53 Data Register (DR28)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	D0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.54 Data Register (DR29)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	D4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.55 Data Register (DR30)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	D8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.56 Data Register (DR31)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	DCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.57 Data Register (DR32)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	E0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.58 Data Register (DR33)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0. NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	E4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.59 Data Register (DR34)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	E8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.60 Data Register (DR35)

The SPI Controller data register is a 16-bit read/write buffer for the transmit/receive FIFOs. When the register is read, data in the receive FIFO buffer is accessed. When it is written to, data is moved into the transmit FIFO buffer; a write can occur only when SSI_EN = 1. FIFOs are reset when SSI_EN = 0.
NOTE : The DR register in the SPI Controller occupies thirty-six 32-bit address locations of the memory map to facilitate AHB burst transfers. Writing to any of these address locations has the same effect as pushing the data from the pwdata bus into the transmit FIFO. Reading from any of these locations has the same effect as retrieving data from the receive FIFO onto the prdata bus. The FIFO buffers on the SPI Controller are not addressable.

MEM Offset (B0001000)	ECh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:16	RO	16'b0	**Reserved 1 (RSVD1)** Reserved

Bits	Access Type	Default	Description

15:0	RW	16'h0	**Data Register (DR)**
			When writing to this register, you must right-justify the data. Read data is automatically right-justified.
			Read = Receive FIFO buffer
			Write = Transmit FIFO buffer

15.3.1.61 RX Sample Delay Register (RX_SAMPLE_DLY)

This register controls the number of ssi_clk cycles that are delayed,from the default sample time,before the actual sample of the rxd input signal occurs. It is impossible to write to this register when the SPI Controller is enabled; the SPI Controller is enabled and disabled by writing to the SSIENR register.

MEM Offset (B0001000)	F0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'h0	**Reserved 1 (RSVD1)** Reserved
3:0	RW/L	4'h0	**Receive Data Sample Delay (RSD)** This register is used to delay the sample of the rxd input signal. Each value represents a single ssi_clk delay on the sample of the rxd signal. NOTE: If this register is programmed with a value that exceeds the depth of the internal shift registers (SSI_RX_DLY_SR_DEPTH

§

16 DMA Controller

The SoC contains a single 2-Channel DMA controller. The DMA controller supports Single or Multi-Block transfers from Memory to Memory, Peripheral to Memory, Memory to Peripheral or Peripheral to Peripheral.

16.1 Features

The following is a list of the DMA Controller features:

- 2 Unidirectional Channels
- Configurable Channel Prioritization
- Software Handshaking
- Hardware Handshaking Interfaces
- Configurable Transfer Type and Flow Control
- Supports Single-Block Transfers
- Supports Multi-Block Transfers
- Scatter-Gather
- Interrupt Generation per Channel:
  - DMA Transfer Complete
  - Block Transfer Complete
  - Source Transaction Complete
  - Destination Transaction Complete
  - Error Response

16.2 Use

DMA Transfers are initiated either through Software or Hardware Handshaking Interfaces. The Handshaking Interface is configurable on a per Channel basis. The following Hardware Handshaking Interfaces are available for selection:

Table 44. Hardware Handshake Interfaces

Interface ID	Peripheral
0	UART 0 TX
1	UART 0 RX
2	UART 1 TX
3	UART 1 RX
4	SPI Master TX
5	SPI Master RX

Interface ID	Peripheral
6	Reserved
7	Reserved
8	SPI Slave TX
9	SPI Slave RX
10	Reserved
11	Reserved
12	I^{2}C Master TX
13	I^{2}C Master RX
14	Reserved
15	Reserved

Transfer Type and Flow Control are configurable on a per Channel basis, the following Flow Control options are available depending on the Transfer Type:

Table 45. Transfer Type and Flow Control Options

Transfer Type	Supported Flow Control Agent(s)
Memory to Memory	DMA Controller
Peripheral to Memory	DMA Controller or Peripheral
Memory to Peripheral	DMA Controller or Peripheral
Peripheral to Peripheral	DMA Controller or Source Peripheral or Destination Peripheral

Multi-Block Transfers are achieved through:

- Linked List (Block Chaining)
- Address Auto-Reloading
- Contiguous Addressing

There are 5 possible sources for Channel Interrupts, each of these sources can be individually masked. The sources are provided in the following list:

1. DMA Transfer Complete - generated on DMA transfer completion to the Destination Peripheral.
2. Block Transfer Complete - generated on DMA block transfer completion to the Destination Peripheral.
3. Source Transaction Complete - generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the source side.

4. Destination Transaction Complete - generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the destination side.
5. Error Response – generated when an ERROR response is received from an AHB slave on the HRESP bus during a DMA transfer.

In the event on an ERROR response the DMA transfer is cancelled and the corresponding channel is disabled, an Error Response interrupt will be generated if the channel configured to do so.

16.3 Memory Mapped IO Registers

Registers listed are for the DMA Controller, starting at base address B0700000h.

MEM Address	Default	Instance Name	Name
0xB0700000	0000_0000h	SAR0	Channel0 Source Address
0xB0700008	0000_0000h	DAR0	Channel0 Destination Address
0xB0700010	0000_0000h	LLP0	Channel0 Linked List Pointer
0xB0700018	0030_4801h	CTL_L0	Channel0 Control LOWER
0xB070001C	0000_0002h	CTL_U0	Channel0 Control UPPER
0xB0700020	0000_0000h	SSTAT0	Channel0 Source Status
0xB0700028	0000_0000h	DSTAT0	Channel0 Destination Status
0xB0700030	0000_0000h	SSTATAR0	Channel0 Source Status Address
0xB0700038	0000_0000h	DSTATAR0	Channel0 Destination Status Address
0xB0700040	0000_0E00h	CFG_L0	Channel0 Configuration LOWER
0xB0700044	0000_0004h	CFG_U0	Channel0 configuration UPPER
0xB0700048	0000_0000h	SGR0	Channel0 Source Gather
0xB0700050	0000_0000h	DSR0	Channel0 Destination Scatter
0xB0700058	0000_0000h	SAR1	Channel1 Source Address
0xB0700060	0000_0000h	DAR1	Channel1 Destination Address
0xB0700068	0000_0000h	LLP1	Channel1 Linked List Pointer
0xB0700070	0030_4801h	CTL_L1	Channel1 Control LOWER
0xB0700074	0000_0002h	CTL_U1	Channel1 Control UPPER
0xB0700078	0000_0000h	SSTAT1	Channel1 Source Status
0xB0700080	0000_0000h	DSTAT1	Channel1 Destination Status
0xB0700088	0000_0000h	SSTATAR1	Channel1 Source Status Address
0xB0700090	0000_0000h	DSTATAR1	Channel1 Destination Status Address
0xB0700098	0000_0E00h	CFG_L1	Channel1 Configuration LOWER
0xB070009C	0000_0004h	CFG_U1	Channel1 configuration UPPER
0xB07000A0	0000_0000h	SGR1	Channel1 Source Gather

MEM Address	Default	Instance Name	Name
0xB07000A8	0000_0000h	DSR1	Channel1 Destination Scatter
0xB07002C0	0000_0000h	RAW_TFR	Raw Status for IntTfr Interrupt
0xB07002C8	0000_0000h	RAW_BLOCK	Raw Status for IntBlock Interrupt
0xB07002D0	0000_0000h	RAW_SRC_TRAN	Raw Status for IntSrcTran Interrupt
0xB07002D8	0000_0000h	RAW_DST_TRAN	Raw Status for IntDstTran Interrupt
0xB07002E0	0000_0000h	RAW_ERR	Raw Status for IntErr Interrupt
0xB07002E8	0000_0000h	STATUS_TFR	Status for IntTfr Interrupt
0xB07002F0	0000_0000h	STATUS_BLOCK	Status for IntBlock Interrupt
0xB07002F8	0000_0000h	STATUS_SRC_TRAN	Status for IntSrcTran Interrupt
0xB0700300	0000_0000h	STATUS_DST_TRAN	Status for IntDstTran Interrupt
0xB0700308	0000_0000h	STATUS_ERR	Status for IntErr Interrupt
0xB0700310	0000_0000h	MASK_TFR	Mask for IntTfr Interrupt
0xB0700318	0000_0000h	MASK_BLOCK	Mask for IntBlock Interrupt
0xB0700320	0000_0000h	MASK_SRC_TRAN	Mask for IntSrcTran Interrupt
0xB0700328	0000_0000h	MASK_DST_TRAN	Mask for IntDstTran Interrupt
0xB0700330	0000_0000h	MASK_ERR	Mask for IntErr Interrupt
0xB0700338	0000_0000h	CLEAR_TFR	Clear for IntTfr Interrupt
0xB0700340	0000_0000h	CLEAR_BLOCK	Clear for IntBlock Interrupt
0xB0700348	0000_0000h	CLEAR_SRC_TRAN	Clear for IntSrcTran Interrupt
0xB0700350	0000_0000h	CLEAR_DST_TRAN	Clear for IntDstTran Interrupt
0xB0700358	0000_0000h	CLEAR_ERR	Clear for IntErr Interrupt
0xB0700360	0000_0000h	STATUS_INT	Combined Interrupt Status
0xB0700368	0000_0000h	REQ_SRC_REG	Source Software Transaction Request
0xB0700370	0000_0000h	REQ_DST_REG	Destination Software Transaction Request register
0xB0700378	0000_0000h	SGL_REQ_SRC_REG	Source Single Transaction Request
0xB0700380	0000_0000h	SGL_REQ_DST_REG	Destination Single Software Transaction Request
0xB0700388	0000_0000h	LST_SRC_REG	Source Last Transaction Request
0xB0700390	0000_0000h	LST_DST_REG	Destination Single Transaction Request
0xB0700398	0000_0000h	DMA_CFG_REG	DMA Configuration
0xB07003A0	0000_0000h	CH_EN_REG	Channel Enable
0xB07003A8	0000_0000h	DMA_ID_REG	DMA ID
0xB07003B0	0000_0000h	DMA_TEST_REG	DMA Test
0xB07003F8	4457_1110h	DMA_COMP_ID_L	DMA Component ID - LOWER
0xB07003FC	3231_382Ah	DMA_COMP_ID_U	DMA Component ID - UPPER

16.3.1.1 Channel0 Source Address (SAR0)

Source Address of DMA transfer
The starting source address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer.
While the DMA transfer is in progress, this register is updated to reflect the source address of the current transfer.

MEM Offset (00000000)	0B0700000h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Current Source Address of DMA transfer (SAR)** Updated after each source transfer. The SINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every source transfer throughout the block transfer.

16.3.1.2 Channel0 Destination Address (DAR0)

Destination address of DMA transfer
The starting destination address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the destination address of the current transfer.

MEM Offset (00000000)	0B0700008h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Current Destination address of DMA transfer (DAR)** Updated after each destination transfer. The DINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every destination transfer throughout the block transfer.

16.3.1.3 Channel0 Linked List Pointer (LLP0)

Program this register to point to the first Linked List Item (LLI) in memory prior to enabling the channel if block chaining is enabled

MEM Offset (00000000)	0B0700010h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RW	30'b0	**Starting Address In Memory (LOC)** Starting Address In Memory of next LLI if block chaining is enabled. Note that the two LSBs of the starting address are not stored because the address is assumed to be aligned to a 32-bit boundary. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
1:0	RO	2'b0	**Reserved (RSV)**

16.3.1.4 Channel0 Control LOWER (CTL_L0)

Contains fields that control the DMA transfer
It is part of the block descriptor (linked list item - LLI) when block chaining is enabled. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled.

MEM Offset (00000000)	0B0700018h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0030_4801h

Bits	Access Type	Default	Description
31:29	RO	3'b0	**Reserved (RSV2)**
28	RW	1'b0	**LLP_SRC_EN (LLP_SRC_EN)** Block chaining is enabled on the source side only if the LLP_SRC_EN field is high and LLPx.LOC is non-zero
27	RW	1'b0	**LLP_DST_EN (LLP_DST_EN)** Block chaining is enabled on the destination side only if the LLP_DST_EN field is high and LLP0.LOC is non-zero
26:25	RO	2'b0	**Source AMBA Layer (SMS)** Hardcoded the Master interface attached to the source of channel 0.
24:23	RO	2'b0	**Destination AMBA Layer (DMS)** Hardcoded the Master interface attached to the destination of channel 0
22:20	RW	3'b011	**Transfer Type and Flow Control (TT_FC)** The following transfer types are supported: *Code - Type - Flow Controller* 000 - Memory to Memory - DMAC

Bits	Access Type	Default	Description
			001 - Memory to Peripheral - DMAC 010 - Peripheral to Memory - DMAC 011 - Peripheral to Peripheral - DMAC 100 - Peripheral to Memory - Peripheral 101 - Peripheral to Peripheral - Source Peripheral 110 - Memory to Peripheral - Peripheral 111 - Peripheral to Peripheral - Destination Peripheral
19	RO	1'b0	**Reserved (RSV0)**
18	RW	1'b0	**Destination scatter enable (DST_SCATTER_EN)** 0 = Scatter disabled 1 = Scatter enabled Scatter on the destination side is applicable only when the CTL0_L.DINC bit indicates an incrementing or decrementing address control.
17	RW	1'b0	**Source gather enable (SRC_GATHER_EN)** 0 = Gather disabled 1 = Gather enabled Gather on the source side is applicable only when the CTL0_L.SINC bit indicates an incrementing or decrementing address control.
16:14	RW	3'b001	**Source Burst Transaction Length (SRC_MSIZE)** Number of data items, each of width CTL0_L.SRC_TR_WIDTH, to be read from the source every time a source burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WIDTH) -------------------- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
13:11	RW	3'b001	**Destination Burst Transaction Length (DEST_MSIZE)** Destination Burst Transaction Length. Number of data items, each of width CTL0_L.DST_TR_WIDTH, to be written to the destination every time a destination burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WIDTH) -------------------- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64

Bits	Access Type	Default	Description
			110 - 128 111 - 256
10:9	RW	2'b0	**Source Address Increment (SINC)** Indicates whether to increment or decrement the source address on every source transfer. If the device is fetching data from a source peripheral FIFO with a fixed address, then set this field to No change. 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next SRC_TR_WIDTH boundary.
8:7	RW	2'b0	**Destination Address Increment (DINC)** Indicates whether to increment or decrement the destination address on every destination transfer. If your device is writing data to a destination peripheral FIFO with a fixed address, then set this field to No change 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next DST_TR_WIDTH boundary.
6:4	RW	3'b000	**Source transfer width (SRC_TR_WIDTH)** Decoding for this field: Value - Size(bits) -------------------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
3:1	RW	3'b000	**Destination transfer width (DST_TR_WIDTH)** Decoding for this field: Value - Size(bits) -------------------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256

Bits	Access Type	Default	Description
0	RW	1'b1	**Interrupt enable (INT_EN)** If set, then all interrupt-generating sources are enabled. Functions as a global mask bit for all interrupts for the channel. RAW interrupt registers still assert if INT_EN = 0.

16.3.1.5 Channel0 Control UPPER (CTL_U0)

Contains fields that control the DMA transfer.
It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled. If status write-back is enabled, the content is written to the control register location of the LLI in system memory at the end of the block transfer.

MEM Offset (00000000)	0B070001Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0002h

Bits	Access Type	Default	Description
31:13	RO	19'b0	**Reserved (RSV1)**
12	RW	1'b0	**Done bit (DONE)** If status write-back is enabled, the upper word of the control register, CTL0_U[31:0], is written to the control register location of the Linked List Item (LLI) in system memory at the end of the block transfer with the done bit set. Software can poll the LLI CTL0.DONE bit to see when a block transfer is complete. The LLI CTL0.DONE bit should be cleared when the linked lists are set up in memory prior to enabling the channel. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
11:0	RW	12'b000000000010	**Block length (BLOCK_TS)** When the DMAC is the flow controller, the user writes this field before the channel is enabled in order to indicate the block size. The number programmed into BLOCK_TS indicates the total number of single transactions to perform for every block transfer. A single transaction is mapped to a single AMBA beat. Width: The width of the single transaction is determined by CTL0_L.SRC_TR_WIDTH.

16.3.1.6 Channel0 Source Status (SSTAT0)

This register is a temporary placeholder for the source status information on its way to the SSTAT0 register location of the LLI. The source status information should be retrieved by software from the SSTAT0 register location of the LLI, and not by a read of this register over the DMAC slave interface.

MEM Offset (00000000)	0B0700020h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Source Status (SSTAT)** Source status information retrieved by hardware from the address pointed to by the contents of the SSTATAR0 register.

16.3.1.7 Channel0 Destination Status (DSTAT0)

This register is a temporary placeholder for the destination status information on its way to the DSTAT0 register location of the LLI. The destination status information should be retrieved by software from the DSTAT0 register location of the LLI and not by a read of this register over the DMAC slave interface

MEM Offset (00000000)	0B0700028h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Destination Status (DSTAT)** Destination status information retrieved by hardware from the address pointed to by the contents of the DSTATAR0 register

16.3.1.8 Channel0 Source Status Address (SSTATAR0)

After the completion of each block transfer, hardware can retrieve the source status information from the address pointed to by the contents of this register

MEM Offset (00000000)	0B0700030h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Source Status Address (SSTATAR)** Pointer from where hardware can fetch the source status information, which is registered in the SSTAT0 register and written out to the SSTAT0 register location of the LLI before the start of the next block.

16.3.1.9 Channel0 Destination Status Address (DSTATAR0)

After the completion of each block transfer, hardware can retrieve the destination status information from the address pointed to by the contents of this register

MEM Offset (00000000)	0B0700038h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Destination Status Address (DSTATAR)** Pointer from where hardware can fetch the destination status information, which is registered in the DSTAT0 register and written out to the DSTAT0 register location of the LLI before the start of the next block.

16.3.1.10 Channel0 Configuration LOWER (CFG_L0)

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. You need to program this register prior to enabling the channel.

MEM Offset (00000000)	0B0700040h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0E00h

Bits	Access Type	Default	Description
31	RW	1'b0	**Reload destination enable (RELOAD_DST)** The DAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
30	RW	1'b0	**Reload source enable (RELOAD_SRC)** The SAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
29:20	RO	10'b0	**Reserved (RSV2)**

Bits	Access Type	Default	Description
19	RW	1'b0	**Source handshake polarity (SRC_HS_POL)** 0 = Active high 1 = Active low Hardware handshake polarity for Peripherals: 1. SPI = Active high 2. I^2C = Active high 3. UART A/B = Active low
18	RW	1'b0	**Destination handshake polarity (DST_HS_POL)** 0 = Active high 1 = Active low Hardware handshake polarity for Peripherals: 1. SPI = Active high 2. I^2C = Active high 3. UART A/B = Active low
17:12	RO	6'b0	**Reserved (RSV1)**
11	RW	1'b1	**Source Handshake select (HS_SEL_SRC)** Used to select which handshake interface is active for source requests on this channel. 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If source peripheral is memory this bit is ignored
10	RW	1'b1	**Destination Handshake select (HS_SEL_DST)** Used to select which handshake interface is active for destination requests on this channel. 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If destination peripheral is memory this bit is ignored
9	RO	1'b1	**Channel FIFO empty status (FIFO_EMPTY)** Indicates if there is data left in the channel FIFO. Can be used in conjunction with CH_SUSP to cleanly disable a channel. 1 = Channel FIFO empty 0 = Channel FIFO not empty
8	RW	1'b0	**Channel Suspend control (CH_SUSP)** Suspends all DMA data transfers from the source until this bit is cleared. There is no guarantee that the current transaction will complete. Can also be used in conjunction with FIFO_EMPTY to cleanly disable a channel without losing any data. 0 = Not suspended. 1 = Suspend DMA transfer from the source.

Bits	Access Type	Default	Description
7:5	RW	3'b0	**Channel Priority (CH_PRIOR)** Priority value equal to 7 is the highest priority, and 0 is the lowest. This field must be programmed within the following range: 0: 1 A programmed value outside this range will cause erroneous behavior.
4:0	RO	5'b0	**Reserved (RSV0)**

16.3.1.11 Channel0 configuration UPPER (CFG_U0)

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. You need to program this register prior to enabling the channel.

MEM Offset (00000000)	0B0700044h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0004h

Bits	Access Type	Default	Description
31:15	RO	17'h0	**Reserved (RSV5)**
14:11	RW	4'h0	**Destination hardware interface (DEST_PER)** Assigns a hardware handshaking interface (0-1) to the channel destination if the CFG0_L.HS_SEL_DST field is 0; otherwise, this field is ignored. The channel can then communicate with the destination peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
10:7	RW	4'h0	**Source hardware interface (SRC_PER)** Assigns a hardware handshaking interface (0-1) to the channel source if the CFG0_L.HS_SEL_SRC field is 0; otherwise, this field is ignored. The channel can then communicate with the source peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
6	RW	1'b0	**Source Status Update Enable (SS_UPD_EN)** Source status information is fetched only from the location pointed to by the SSTATAR0 register, stored in the SSTAT0 register and written out to the SSTAT0 location of the LLI if this field is high
5	RW	1'b0	**Destination Status Update Enable (DS_UPD_EN)** Destination status information is fetched only from the location pointed to by the DSTATAR0 register, stored in the DSTAT0 register and written out to the DSTAT0 location of the LLI if this field is high

Bits	Access Type	Default	Description
4:2	RW	3'b001	**AHB bus protocol bus control (PROTCTL)** Protection Control bits used to drive the AHB HPROT[3:1] bus. The AMBA Specification recommends that the default value of HPROT indicates a non-cached, non-buffered, privileged data access. The reset value is used to indicate such an access. HPROT[0] is tied high because all transfers are data accesses, as there are no opcode fetches. There is a one-to-one mapping of these register bits to the HPROT[3:1] master interface signals.
1	RW	1'b0	**Channel FIFO mode control (FIFO_MODE)** Determines how much space or data needs to be available in the FIFO before a burst transaction request is serviced. 0 = Space/data available for single AHB transfer of the specified transfer width. 1 = Data available is greater than or equal to half the FIFO depth for destination transfers and space available is greater than half the fifo depth for source transfers. The exceptions are at the end of a burst transaction request or at the end of a block transfer.
0	RW	1'b0	**Channel flow control mode (FCMODE)** Determines when source transaction requests are serviced when the Destination Peripheral is the flow controller. 0 = Source transaction requests are serviced when they occur. Data pre-fetching is enabled. 1 = Source transaction requests are not serviced until a destination transaction request occurs. In this mode, the amount of data transferred from the source is limited so that it is guaranteed to be transferred to the destination prior to block termination by the destination. Data pre-fetching is disabled.

16.3.1.12 Channel0 Source Gather (SGR0)

The CTL0_L.SINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

MEM Offset (00000000)	0B0700048h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	PowerWell	ResetSignal
31:25	RO	7'b0	**Reserved (RSV)**		
24:20	RW	5'b0	**Source Gather Count (SGC)** Source contiguous transfer count between successive gather boundaries. Specifies the number of contiguous source transfers of CTL0_L.SRC_TR_WIDTH between successive gather intervals. This is defined as a gather boundary		

Bits	Access Type	Default	Description	PowerWell	ResetSignal
19:0	RW	20'b0	**Source Gather Interval (SGI)** Specifies the source address increment/decrement in multiples of CTL0_L.SRC_TR_WIDTH on a gather boundary when gather mode is enabled for the source transfer.		

16.3.1.13 Channel0 Destination Scatter (DSR0)

The CTL0_L.DINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

MEM Offset (00000000)	0B0700050h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	7'b0	**Reserved (RSV)**
24:20	RW	5'b0	**Destination Scatter Count (DSC)** Source contiguous transfer count between successive scatter boundaries. Specifies the number of contiguous destination transfers of CTL0_L.DST_TR_WIDTH between successive scatter intervals. This is defined as a scatter boundary
19:0	RW	20'b0	**Destination Scatter Interval (DSI)** Specifies the destination address increment/decrement in multiples of CTL0_L.DST_TR_WIDTH on a scatter boundary when scatter mode is enabled for the destination transfer.

16.3.1.14 Channel1 Source Address (SAR1)

Source Address of DMA transfer
The starting source address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer.
While the DMA transfer is in progress, this register is updated to reflect the source address of the current transfer.

MEM Offset (00000000)	0B0700058h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Current Source Address of DMA transfer (SAR)** Updated after each source transfer. The SINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every source transfer throughout the block transfer.

16.3.1.15 Channel1 Destination Address (DAR1)

Destination address of DMA transfer.
The starting destination address is programmed by software before the DMA channel is enabled, or by an LLI update before the start of the DMA transfer. While the DMA transfer is in progress, this register is updated to reflect the destination address of the current transfer.

MEM Offset (00000000)	0B0700060h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Current Destination address of DMA transfer (DAR)** Updated after each destination transfer. The DINC field in the CTL0_L register determines whether the address increments, decrements, or is left unchanged on every destination transfer throughout the block transfer.

16.3.1.16 Channel1 Linked List Pointer (LLP1)

Program this register to point to the first Linked List Item (LLI) in memory prior to enabling the channel if block chaining is enabled

MEM Offset (00000000)	0B0700068h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RW	30'b0	**Starting Address In Memory (LOC)** Starting Address In Memory of next LLI if block chaining is enabled. Note that the two LSBs of the starting address are not stored because the address is assumed to be aligned to a 32-bit boundary. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
1:0	RO	2'b0	**Reserved (RSV)**

16.3.1.17 Channel1 Control LOWER (CTL_L1)

Contains fields that control the DMA transfer
It is part of the block descriptor (linked list item - LLI) when block chaining is enabled. It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled.

MEM Offset (00000000)	0B0700070h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0030_4801h

Bits	Access Type	Default	Description
31:29	RO	3'b0	**Reserved (RSV2)**
28	RW	1'b0	**LLP_SRC_EN (LLP_SRC_EN)** Block chaining is enabled on the source side only if the LLP_SRC_EN field is high and LLPx.LOC is non-zero
27	RW	1'b0	**LLP_DST_EN (LLP_DST_EN)** Block chaining is enabled on the destination side only if the LLP_DST_EN field is high and LLP0.LOC is non-zero
26:25	RO	2'b0	**Source AMBA Layer (SMS)** Hardcoded the Master interface attached to the source of channel 0.
24:23	RO	2'b0	**Destination AMBA Layer (DMS)** Hardcoded the Master interface attached to the destination of channel 0
22:20	RW	3'b011	**Transfer Type and Flow Control (TT_FC)** The following transfer types are supported: *Code - Type - Flow Controller* 000 - Memory to Memory - DMAC 001 - Memory to Peripheral - DMAC 010 - Peripheral to Memory - DMAC 011 - Peripheral to Peripheral - DMAC 100 - Peripheral to Memory - Peripheral 101 - Peripheral to Peripheral - Source Peripheral 110 - Memory to Peripheral - Peripheral 111 - Peripheral to Peripheral - Destination Peripheral
19	RO	1'b0	**Reserved (RSV0)**
18	RW	1'b0	**Destination scatter enable (DST_SCATTER_EN)** 0 = Scatter disabled 1 = Scatter enabled Scatter on the destination side is applicable only when the CTL0_L.DINC bit indicates an incrementing or decrementing address control.

Bits	Access Type	Default	Description
17	RW	1'b0	**Source gather enable (SRC_GATHER_EN)** 0 = Gather disabled 1 = Gather enabled Gather on the source side is applicable only when the CTL0_L.SINC bit indicates an incrementing or decrementing address control.
16:14	RW	3'b001	**Source Burst Transaction Length (SRC_MSIZE)** Number of data items, each of width CTL0_L.SRC_TR_WIDTH, to be read from the source every time a source burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WIDTH) -------------------- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
13:11	RW	3'b001	**Destination Burst Transaction Length (DEST_MSIZE)** Destination Burst Transaction Length. Number of data items, each of width CTL0_L.DST_TR_WIDTH, to be written to the destination every time a destination burst transaction request is made from either the corresponding hardware or software handshaking interface. Value - Size(#TR_WIDTH) -------------------- 000 - 1 001 - 4 010 - 8 011 - 16 100 - 32 101 - 64 110 - 128 111 - 256
10:9	RW	2'b0	**Source Address Increment (SINC)** Indicates whether to increment or decrement the source address on every source transfer. If the device is fetching data from a source peripheral FIFO with a fixed address, then set this field to No change. 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next SRC_TR_WIDTH boundary.

Bits	Access Type	Default	Description
8:7	RW	2'b0	**Destination Address Increment (DINC)** Indicates whether to increment or decrement the destination address on every destination transfer. If your device is writing data to a destination peripheral FIFO with a fixed address, then set this field to No change 00 = Increment 01 = Decrement 1x = No change NOTE: Incrementing or decrementing is done for alignment to the next DST_TR_WIDTH boundary.
6:4	RW	3'b000	**Source transfer width (SRC_TR_WIDTH)** Decoding for this field: Value - Size(bits) -------------------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
3:1	RW	3'b000	**Destination transfer width (DST_TR_WIDTH)** Decoding for this field: Value - Size(bits) -------------------- 000 - 8 001 - 16 010 - 32 011 - 64 100 - 128 101 - 256 11x - 256
0	RW	1'b1	**Interrupt enable (INT_EN)** If set, then all interrupt-generating sources are enabled. Functions as a global mask bit for all interrupts for the channel. RAW interrupt registers still assert if INT_EN = 0.

16.3.1.18 Channel1 Control UPPER (CTL_U1)

Contains fields that control the DMA transfer.
It can be varied on a block-by-block basis within a DMA transfer when block chaining is enabled. If status write-back is enabled, the content is written to the control register location of the LLI in system memory at the end of the block transfer.

MEM Offset (00000000)	0B0700074h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0002h

Bits	Access Type	Default	Description
31:13	RO	19'b0	**Reserved (RSV1)**
12	RW	1'b0	**Done bit (DONE)** If status write-back is enabled, the upper word of the control register, CTL0_U[31:0], is written to the control register location of the Linked List Item (LLI) in system memory at the end of the block transfer with the done bit set. Software can poll the LLI CTL0.DONE bit to see when a block transfer is complete. The LLI CTL0.DONE bit should be cleared when the linked lists are set up in memory prior to enabling the channel. LLI accesses are always 32-bit accesses (Hsize = 2) aligned to 32-bit boundaries and cannot be changed or programmed to anything other than 32-bit.
11:0	RW	12'b000000000010	**Block length (BLOCK_TS)** When the DMAC is the flow controller, the user writes this field before the channel is enabled in order to indicate the block size. The number programmed into BLOCK_TS indicates the total number of single transactions to perform for every block transfer. A single transaction is mapped to a single AMBA beat. Width: The width of the single transaction is determined by CTL0_L.SRC_TR_WIDTH.

16.3.1.19 Channel1 Source Status (SSTAT1)

This register is a temporary placeholder for the source status information on its way to the SSTAT0 register location of the LLI. The source status information should be retrieved by software from the SSTAT0 register location of the LLI, and not by a read of this register over the DMAC slave interface.

MEM Offset (00000000)	0B0700078h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Source Status (SSTAT)** Source status information retrieved by hardware from the address pointed to by the contents of the SSTATAR0 register.

16.3.1.20 Channel1 Destination Status (DSTAT1)

This register is a temporary placeholder for the destination status information on its way to the DSTAT0 register location of the LLI. The destination status information should be retrieved by software from the DSTAT0 register location of the LLI and not by a read of this register over the DMAC slave interface

MEM Offset (00000000)	0B0700080h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Destination Status (DSTAT)** Destination status information retrieved by hardware from the address pointed to by the contents of the DSTATAR0 register

16.3.1.21 Channel1 Source Status Address (SSTATAR1)

After the completion of each block transfer, hardware can retrieve the source status information from the address pointed to by the contents of this register

MEM Offset (00000000)	0B0700088h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Source Status Address (SSTATAR)** Pointer from where hardware can fetch the source status information, which is registered in the SSTAT0 register and written out to the SSTAT0 register location of the LLI before the start of the next block.

16.3.1.22 Channel1 Destination Status Address (DSTATAR1)

After the completion of each block transfer, hardware can retrieve the destination status information from the address pointed to by the contents of this register

MEM Offset (00000000)	0B0700090h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'b0	**Channel Destination Status Address (DSTATAR)** Pointer from where hardware can fetch the destination status information, which is registered in the DSTAT0 register and written out to the DSTAT0 register location of the LLI before the start of the next block.

16.3.1.23 Channel1 Configuration LOWER (CFG_L1)

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. You need to program this register prior to enabling the channel.

MEM Offset (00000000)	0B0700098h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0E00h

Bits	Access Type	Default	Description
31	RW	1'b0	**Reload destination enable (RELOAD_DST)** The DAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
30	RW	1'b0	**Reload source enable (RELOAD_SRC)** The SAR register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated.
29:20	RO	10'b0	**Reserved (RSV2)**
19	RW	1'b0	**Source handshake polarity (SRC_HS_POL)** 0 = Active high 1 = Active low Hardware handshake polarity for Peripherals: 1. SPI = Active high 2. I^{2}C = Active high 3. UART A/B = Active low
18	RW	1'b0	**Destination handshake polarity (DST_HS_POL)** 0 = Active high 1 = Active low Hardware handshake polarity for Peripherals: 1. SPI = Active high 2. I^{2}C = Active high 3. UART A/B = Active low
17:12	RO	6'b0	**Reserved (RSV1)**

Bits	Access Type	Default	Description
11	RW	1'b1	**Source Handshake select (HS_SEL_SRC)** Used to select which handshake interface is active for source requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If source peripheral is memory this bit is ignored
10	RW	1'b1	**Destination Handshake select (HS_SEL_DST)** Used to select which handshake interface is active for destination requests on this channel 0 = HW handshake. SW ones are ignored 1 = SW handshake. HW ones are ignored If destination peripheral is memory this bit is ignored
9	RO	1'b1	**Channel FIFO empty status (FIFO_EMPTY)** Indicates if there is data left in the channel FIFO. Can be used in conjunction with CH_SUSP to cleanly disable a channel. 1 = Channel FIFO empty 0 = Channel FIFO not empty
8	RW	1'b0	**Channel Suspend control (CH_SUSP)** Suspends all DMA data transfers from the source until this bit is cleared. There is no guarantee that the current transaction will complete. Can also be used in conjunction with FIFO_EMPTY to cleanly disable a channel without losing any data. 0 = Not suspended. 1 = Suspend DMA transfer from the source.
7:5	RW	3'b0	**Channel Priority (CH_PRIOR)** Priority value equal to 7 is the highest priority, and 0 is the lowest. This field must be programmed within the following range: 0: 1 A programmed value outside this range will cause erroneous behavior.
4:0	RO	5'b0	**Reserved (RSV0)**

16.3.1.24 Channel1 configuration UPPER (CFG_U1)

Contains fields that configure the DMA transfer. The channel configuration register remains fixed for all blocks of a multi-block transfer. You need to program this register prior to enabling the channel.

MEM Offset (00000000)	0B070009Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0004h

Bits	Access Type	Default	Description
31:15	RO	17'h0	**Reserved (RSV5)**
14:11	RW	4'h0	**Destination hardware interface (DEST_PER)** Assigns a hardware handshaking interface (0-1) to the channel destination if the CFG0_L.HS_SEL_DST field is 0; otherwise, this field is ignored. The channel can then communicate with the destination peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
10:7	RW	4'h0	**Source hardware interface (SRC_PER)** Assigns a hardware handshaking interface (0-1) to the channel source if the CFG0_L.HS_SEL_SRC field is 0; otherwise, this field is ignored. The channel can then communicate with the source peripheral connected to that interface through the assigned hardware handshaking interface. NOTE: For correct DMA operation, only one peripheral (source or destination) should be assigned to the same handshaking interface.
6	RW	1'b0	**Source Status Update Enable (SS_UPD_EN)** Source status information is fetched only from the location pointed to by the SSTATAR0 register, stored in the SSTAT0 register and written out to the SSTAT0 location of the LLI if this field is high
5	RW	1'b0	**Destination Status Update Enable (DS_UPD_EN)** Destination status information is fetched only from the location pointed to by the DSTATAR0 register, stored in the DSTAT0 register and written out to the DSTAT0 location of the LLI if this field is high
4:2	RW	3'b001	**AHB bus protocol bus control (PROTCTL)** Protection Control bits used to drive the AHB HPROT[3:1] bus. The AMBA Specification recommends that the default value of HPROT indicates a non-cached, non-buffered, privileged data access. The reset value is used to indicate such an access. HPROT[0] is tied high because all transfers are data accesses, as there are no opcode fetches. There is a one-to-one mapping of these register bits to the HPROT[3:1] master interface signals.
1	RW	1'b0	**Channel FIFO mode control (FIFO_MODE)** Determines how much space or data needs to be available in the FIFO before a burst transaction request is serviced. 0 = Space/data available for single AHB transfer of the specified transfer width. 1 = Data available is greater than or equal to half the FIFO depth for destination transfers and space available is greater than half the fifo depth for source transfers. The exceptions are at the end of a burst transaction request or at the end of a block transfer.

Bits	Access Type	Default	Description
0	RW	1'b0	**Channel flow control mode (FCMODE)** Determines when source transaction requests are serviced when the Destination Peripheral is the flow controller. 0 = Source transaction requests are serviced when they occur. Data pre-fetching is enabled. 1 = Source transaction requests are not serviced until a destination transaction request occurs. In this mode, the amount of data transferred from the source is limited so that it is guaranteed to be transferred to the destination prior to block termination by the destination. Data pre-fetching is disabled.

16.3.1.25 Channel1 Source Gather (SGR1)

The CTL0_L.SINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

MEM Offset (00000000)	0B07000A0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	7'b0	**Reserved (RSV)**
24:20	RW	5'b0	**Source Gather Count (SGC)** Source contiguous transfer count between successive gather boundaries. Specifies the number of contiguous source transfers of CTL0_L.SRC_TR_WIDTH between successive gather intervals. This is defined as a gather boundary
19:0	RW	20'b0	**Source Gather Interval (SGI)** Specifies the source address increment/decrement in multiples of CTL0_L.SRC_TR_WIDTH on a gather boundary when gather mode is enabled for the source transfer.

16.3.1.26 Channel1 Destination Scatter (DSR1)

The CTL0_L.DINC field controls whether the address increments or decrements. For a fixed-address control, then the address remains constant throughout the transfer and this register is ignored.

MEM Offset (00000000)	0B07000A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	7'b0	**Reserved (RSV)**
24:20	RW	5'b0	**Destination Scatter Count (DSC)** Source contiguous transfer count between successive scatter boundaries. Specifies the number of contiguous destination transfers of CTL0_L.DST_TR_WIDTH between successive scatter intervals. This is defined as a scatter boundary
19:0	RW	20'b0	**Destination Scatter Interval (DSI)** Specifies the destination address increment/decrement in multiples of CTL0_L.DST_TR_WIDTH on a scatter boundary when scatter mode is enabled for the destination transfer.

16.3.1.27 Raw Status for IntTfr Interrupt (RAW_TFR)

DMA Transfer Complete Interrupt. This interrupt is generated on DMA transfer completion to the destination peripheral

MEM Offset (00000000)	0B07002C0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RW	2'b0	**Raw Status for IntTfr Interrupt (RAW)** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

16.3.1.28 Raw Status for IntBlock Interrupt (RAW_BLOCK)

Block Transfer Complete Interrupt. This interrupt is generated on DMA block transfer completion to the destination peripheral.

MEM Offset (00000000)	0B07002C8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RW	2'b0	**Raw Status for IntBlock Interrupt (RAW)** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

16.3.1.29 Raw Status for IntSrcTran Interrupt (RAW_SRC_TRAN)

Source Transaction Complete Interrupt. Generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the source side. NOTE: If the source is memory, then IntSrcTran interrupt should be ignored, as there is no concept of a DMA transaction level for memory

MEM Offset (00000000)	0B07002D0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RW	2'b0	**Raw Status for IntSrcTran Interrupt (RAW)** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

16.3.1.30 Raw Status for IntDstTran Interrupt (RAW_DST_TRAN)

Destination Transaction Complete Interrupt. Generated after completion of the last AHB transfer of the requested single/burst transaction from the handshaking interface (either the hardware or software handshaking interface) on the destination side. NOTE: If the destination for a channel is memory, then that channel will never generate the IntDstTran interrupt. Because of this, the corresponding bit in this field will not be set.

MEM Offset (00000000)	0B07002D8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RW	2'b0	**Raw Status for IntDstTran Interrupt (RAW)** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the correspondent Clear register

16.3.1.31 Raw Status for IntErr Interrupt (RAW_ERR)

Error Interrupt. This interrupt is generated when an ERROR response is received from an AHB slave on the HRESP bus during a DMA transfer. In addition, the DMA transfer is cancelled and the channel is disabled.

Peripheral controllers (SPI/I^{2}C/UART) or memory controllers don’t generate error response. Only AHB Fabric can generate error response if address points to a hole. DMAC doesn't support slave AHB error response detection for Channel0 source and Channel1 destination transfers, which is acceptable limitation as only AHB fabric can generate error response and that too for address hole.

MEM Offset (00000000)	0B07002E0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RW	2'b0	**Raw Status for IntErr Interrupt (RAW)** Interrupt events are stored in this Raw Interrupt Status register before masking. Each bit in this register is cleared by writing a 1 to the corresponding location in the corresponding Clear register.

16.3.1.32 Status for IntTfr Interrupt (STATUS_TFR)

DMA Transfer Complete Interrupt status

MEM Offset (00000000)	0B07002E8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RO	2'b0	**Status for IntTfr Interrupt (STATUS)** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

16.3.1.33 Status for IntBlock Interrupt (STATUS_BLOCK)

Block Transfer Complete Interrupt status

MEM Offset (00000000)	0B07002F0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RO	2'b0	**Status for IntBlock Interrupt (STATUS)** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

16.3.1.34 Status for IntSrcTran Interrupt (STATUS_SRC_TRAN)

Source Transaction Complete Interrupt status

MEM Offset (00000000)	0B07002F8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RO	2'b0	**Status for IntSrcTran Interrupt (STATUS)** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

16.3.1.35 Status for IntDstTran Interrupt (STATUS_DST_TRAN)

Destination Transaction Complete Interrupt status

MEM Offset (00000000) 0B0700300h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RO	2'b0	**Status for IntDstTran Interrupt (STATUS)** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals

16.3.1.36 Status for IntErr Interrupt (STATUS_ERR)

Error Interrupt status

MEM Offset (00000000) 0B0700308h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	RO	2'b0	**Status for IntErr Interrupt (STATUS)** Stores all interrupt events from channels after masking. One bit allocated per channel. Used to generate the DMAC interrupt signals.

16.3.1.37 Mask for IntTfr Interrupt (MASK_TFR)

DMA Transfer Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

MEM Offset (00000000) 0B0700310h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV2)**

Bits	Access Type	Default	Description
9:8	RW	2'b0	**Interrupt Mask Write Enable (INT_MASK_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV1)**
1:0	RW	2'b0	**Mask for the interrupt (INT_MASK)** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

16.3.1.38 Mask for IntBlock Interrupt (MASK_BLOCK)

Block Transfer Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

MEM Offset (00000000)	0B0700318h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV2)**
9:8	RW	2'b0	**Interrupt Mask Write Enable (INT_MASK_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV1)**
1:0	RW	2'b0	**Mask for the interrupt (INT_MASK)** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

16.3.1.39 Mask for IntSrcTran Interrupt (MASK_SRC_TRAN)

Source Transaction Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

MEM Offset (00000000)	0B0700320h
Security_PolicyGroup	
IntelRsvd	False

Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV2)**
9:8	RW	2'b0	**Interrupt Mask Write Enable (INT_MASK_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV1)**
1:0	RW	2'b0	**Mask for the interrupt (INT_MASK)** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

16.3.1.40 Mask for IntDstTran Interrupt (MASK_DST_TRAN)

Destination Transaction Complete Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

MEM Offset (00000000) 0B0700328h
Security_PolicyGroup
IntelRsvd False
Size 32 bits
Default 0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV2)**
9:8	RW	2'b0	**Interrupt Mask Write Enable (INT_MASK_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV1)**
1:0	RW	2'b0	**Mask for the interrupt (INT_MASK)** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

16.3.1.41 Mask for IntErr Interrupt (MASK_ERR)

Error Interrupt mask. The contents of the Raw Status register is masked with the contents of the Mask register.

MEM Offset (00000000)	0B0700330h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV2)**
9:8	RW	2'b0	**Interrupt Mask Write Enable (INT_MASK_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV1)**
1:0	RW	2'b0	**Mask for the interrupt (INT_MASK)** Written only if the corresponding mask write enable bit in the INT_MASK_WE field is asserted on the same AHB write transfer. This allows software to set a mask bit without performing a read-modified write operation. 0 = masked 1 = unmasked

16.3.1.42 Clear for IntTfr Interrupt (CLEAR_TFR)

DMA Transfer Complete Interrupt clear

MEM Offset (00000000)	0B0700338h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	WO	2'b0	**Clear for Interrupt (CLEAR)** 0 = no effect 1 = clear interrupt

16.3.1.43 Clear for IntBlock Interrupt (CLEAR_BLOCK)

Block Transfer Complete Interrupt clear

MEM Offset (00000000)	0B0700340h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	WO	2'b0	**Clear for Interrupt (CLEAR)** 0 = no effect 1 = clear interrupt

16.3.1.44 Clear for IntSrcTran Interrupt (CLEAR_SRC_TRAN)

Source Transaction Complete Interrupt clear

MEM Offset (00000000)	0B0700348h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	WO	2'b0	**Clear for Interrupt (CLEAR)** 0 = no effect 1 = clear interrupt

16.3.1.45 Clear for IntDstTran Interrupt (CLEAR_DST_TRAN)

Destination Transaction Complete Interrupt clear

MEM Offset (00000000)	0B0700350h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	WO	2'b0	**Clear for Interrupt (CLEAR)** 0 = no effect 1 = clear interrupt

16.3.1.46 Clear for IntErr Interrupt (CLEAR_ERR)

Error Interrupt clear

MEM Offset (00000000)	0B0700358h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'b0	**Reserved (RSV)**
1:0	WO	2'b0	**Clear for Interrupt (STATUS)** 0 = no effect 1 = clear interrupt

16.3.1.47 Combined Interrupt Status (STATUS_INT)

The contents of each of the Status registers is ORed to produce a single bit for each interrupt type in this Combined Interrupt Status register.

MEM Offset (00000000)	0B0700360h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:5	RO	27'b0	**Reserved (RSV)**
4	RO	1'b0	**OR of the contents of STATUS_ERR (ERR)**
3	RO	1'b0	**OR of the contents of STATUS_DSTT (DSTT)**
2	RO	1'b0	**OR of the contents of STATUS_SRCT (SRCT)**
1	RO	1'b0	**OR of the contents of STATUS_BLOCK (BLOCK)**
0	RO	1'b0	**OR of the contents of STATUS_TFR (TFR)**

16.3.1.48 Source Software Transaction Request (REQ_SRC_REG)

MEM Offset (00000000)	0B0700368h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Source Software Transaction Request write enable (SRC_REQ_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Source Software Transaction Request register (SRC_REQ)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.49 Destination Software Transaction Request register (REQ_DST_REG)

MEM Offset (00000000)	0B0700370h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Destination Software Transaction Request write enable (DST_REQ_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Destination Transaction Request register (DST_REQ)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.50 Source Single Transaction Request (SGL_REQ_SRC_REG)

MEM Offset (00000000)	0B0700378h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Source Single Transaction Request write enable (SRC_SGLREQ_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Source Single Transaction Request register (SRC_SGLREQ)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.51 Destination Single Software Transaction Request (SGL_REQ_DST_REG)

MEM Offset (00000000)	0B0700380h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Destination Single Transaction Request write enable (DST_SGLREQ_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Destination Single Transaction Request register (DST_SGLREQ)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.52 Source Last Transaction Request (LST_SRC_REG)

MEM Offset (00000000)	0B0700388h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Source Last Transaction Request write enable (LSTSRC_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Source Last Transaction Request register (LSTSRC)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.53 Destination Single Transaction Request (LST_DST_REG)

MEM Offset (00000000)	0B0700390h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	RW	2'b0	**Destination Last Transaction Request write enable (LSTDST_WE)** 0 = write disabled 1 = write enabled
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Destination Last Transaction Request register (LSTDST)** This bit is written only if the corresponding channel write enable bit in the Write Enable field is asserted on the same AHB write transfer, and if the channel is enabled in the CH_EN_REG register

16.3.1.54 DMA Configuration (DMA_CFG_REG)

Used to enable the DMA controller (DMAC), which must be done before any channel activity can begin. If the global channel enable bit is cleared while any channel is still active, then DMA_EN still returns 1 to indicate that there are channels still active until hardware has terminated all activity on all channels, at which point the DMA_EN bit returns 0

MEM Offset (00000000)	0B0700398h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'b0	**DMA global enable (DMA_EN)** 0 = disabled 1 = enabled

16.3.1.55 Channel Enable (CH_EN_REG)

Software can read this register in order to find out which channels are currently inactive if needs to set up a new channel. It can then enable an inactive channel with the required priority.

MEM Offset (00000000)	0B07003A0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:10	RO	22'b0	**Reserved (RSV1)**
9:8	WO	2'b0	**Channel enable register (CH_EN_WE)**
7:2	RO	6'b0	**Reserved (RSV0)**
1:0	RW	2'b0	**Channel enable register (CH_EN)** Setting this bit enables a channel. Clearing this bit disables the channel. 0 = Disable the Channel 1 = Enable the Channel The CH_EN_REG.CH_EN bit is automatically cleared by hardware to disable the channel after the last AMBA transfer of the DMA transfer to the destination has completed. Software can therefore poll this bit to determine when this channel is free for a new DMA transfer.

16.3.1.56 DMA ID (DMA_ID_REG)

Reads back the coreConsultant configured hardcoded ID number

MEM Offset (00000000)	0B07003A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO	32'b0	**DMA ID (DMA_ID)** Hardcoded ID number

16.3.1.57 DMA Test (DMA_TEST_REG)

This register is used to put the AHB slave interface into test mode, during which the readback value of the writable registers match the value written. In normal operation, the readback value of some registers is a function of the DMA state and does not match the value written.

MEM Offset (00000000)	0B07003B0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'b0	**DMA Test register (TEST_SLV_IF)** Puts the AHB slave interface into test mode. In this mode, the readback value of the writable registers always matches the value written. This bit does not allow writing to read-only registers. 0 = Normal mode 1 = Test mode

16.3.1.58 DMA Component ID - LOWER (DMA_COMP_ID_L)

Read-only register that specifies the version of the packaged component

MEM Offset (00000000)	0B07003F8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	4457_1110h

Bits	Access Type	Default	Description
31:0	RO	32'h44571110	**DMA Component Version (DMA_COMP_TYPE)** Version of the packaged component

16.3.1.59 DMA Component ID - UPPER (DMA_COMP_ID_U)

Read-only register that specifies the component type

MEM Offset (00000000)	0B07003FCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	3231_382Ah

Bits	Access Type	Default	Description
31:0	RO	32'h3231382A	**DMA Component identifier (DMA_COMP_VERSION)** Fixed value

§

17 General Purpose I/O (GPIO)

The SoC contains a single instance of the GPIO controller. The GPIO controller provides a total of 25 GPIOs.

17.1 Signal Descriptions

Please see Chapter 2, “Physical Interfaces” for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, “Electrical Characteristics”
- **Description:** A brief explanation of the signal’s function

Table 46. Memory Signals

Signal Name	Direction/ Type	Description
GPIO[24:0]	I/O /IOVDD	**General Purpose IO:** *25 General Purpose IOs*

17.2 Features

The following is a list of the GPIO controller features:

- 25 independently configurable GPIOs
- Separate data register bit and data direction control bit for each GPIO
- Metastability registers for GPIO read data
- Interrupt mode supported for all GPIOs, configurable as follows:
  - Active High Level
  - Active Low Level
  - Rising Edge
  - Falling Edge
  - Both Edge
- Debounce logic for interrupt sources

17.3 Memory Mapped IO Registers

Registers listed are for GPIO, starting at base address B0000C00h.

Table 47. Summary of GPIO Registers—0xB0000C00

MEM Address	Default	Instance Name	Name
0xB0000C00	0000_0000h	GPIO_SWPORTA_DR	Port A Data
0xB0000C04	0000_0000h	GPIO_SWPORTA_DDR	Port A Data Direction
0xB0000C08	0000_0000h	GPIO_SWPORTA_CTL	Port A Data Source
0xB0000C30	0000_0000h	GPIO_INTEN	Interrupt Enable
0xB0000C34	0000_0000h	GPIO_INTMASK	Interrupt Mask
0xB0000C38	0000_0000h	GPIO_INTTYPE_LEVEL	Interrupt Type
0xB0000C3C	0000_0000h	GPIO_INT_POLARITY	Interrupt Polarity
0xB0000C40	0000_0000h	GPIO_INTSTATUS	Interrupt Status
0xB0000C44	0000_0000h	GPIO_RAW_INTSTATUS	Raw Interrupt Status
0xB0000C48	0000_0000h	GPIO_DEBOUNCE	Debounce Enable
0xB0000C4C	0000_0000h	GPIO_PORTA_EOI	Clear Interrupt
0xB0000C50	0000_0000h	GPIO_EXT_PORTA	Port A External Port
0xB0000C60	0000_0000h	GPIO_LS_SYNC	Synchronization Level
0xB0000C68	0000_0000h	GPIO_INT_BOTHEDGE	Interrupt both edge type
0xB0000C70	0003_9C73h	GPIO_CONFIG_REG2	GPIO Configuration Register 2
0xB0000C74	001F_70F6h	GPIO_CONFIG_REG1	GPIO Configuration Register 1

17.3.1.1 Port A Data (GPIO_SWPORTA_DR)

Contains the GPIO Port data bits

MEM Offset (B0000C00)	0B0000C00h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Port Data (GPIO_SWPORTA_DR)** Values written to this register are output on the I/O signals for if the corresponding data direction bits are set to Output mode and the corresponding control bit for the Port is set to Software mode. The value read back is equal to the last value written to this register

17.3.1.2 Port A Data Direction (GPIO_SWPORTA_DDR)

Used to control the GPIO Port bits data direction

MEM Offset (B0000C00)	0B0000C04h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Port Data Direction (GPIO_SWPORTA_DDR)** Values written to this register independently control the direction of the corresponding data bit in the Port - 0 Input (default) - 1 Output

17.3.1.3 Port A Data Source (GPIO_SWPORTA_CTL)

Used to control the GPIO Port Data Source

MEM Offset (B0000C00)	0B0000C08h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved
0	RW	1'b0	**RSVD (RSVD)** Reserved

17.3.1.4 Interrupt Enable (GPIO_INTEN)

Used to configured Port A bits as interrupt sources

MEM Offset (B0000C00)	0B0000C30h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Interrupt Enable (GPIO_INTEN)**

Bits	Access Type	Default	Description
			Allows each bit of Port A to be configured for interrupts. By default the generation of interrupts is disabled. Whenever a 1 is written to a bit of this register, it configures the corresponding bit on Port A to become an interrupt; otherwise, Port A operates as a normal GPIO signal. Interrupts are disabled on the corresponding bits of Port A if the corresponding data direction register is set to Output. 0 Configure Port A bit as normal GPIO signal (default) 1 Configure Port A bit as interrupt

17.3.1.5 Interrupt Mask (GPIO_INTMASK)

Controls masking for Port A bits configured as interrupt sources

MEM Offset (B0000C00)	0B0000C34h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Interrupt Mask (GPIO_INTMASK)** Controls whether an interrupt on Port A can create an interrupt for the interrupt controller by not masking it. By default, all interrupts bits are unmasked. Whenever a 1 is written to a bit in this register, it masks the interrupt generation capability for this signal; otherwise interrupts are allowed through. The unmasked status can be read as well as the resultant status after masking. 0 Interrupt bits are unmasked (default) 1 Mask interrupt

17.3.1.6 Interrupt Type (GPIO_INTTYPE_LEVEL)

Controls the type of interrupt associated with Port A bits configured as interrupt source

MEM Offset (B0000C00)	0B0000C38h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Interrupt Type (GPIO_INTYPE_LEVEL)**

Bits	Access Type	Default	Description
			Controls the type of interrupt that can occur on Port A. Whenever a 0 is written to a bit of this register, it configures the interrupt type to be level-sensitive; otherwise, it is edge-sensitive. 0 Level-sensitive (default) 1 Edge-sensitive

17.3.1.7 Interrupt Polarity (GPIO_INT_POLARITY)

Controls the interrupt polarity associated with Port A bits configured as interrupt sources

MEM Offset (B0000C00)	0B0000C3Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Interrupt Polarity (GPIO_INT_POLARITY)** Controls the polarity of edge or level sensitivity that can occur on input of Port A. Whenever a 0 is written to a bit of this register, it configures the interrupt type to falling-edge or active-low sensitive; otherwise, it is rising-edge or active-high sensitive. 0 Active-low (default) 1 Active-high

17.3.1.8 Interrupt Status (GPIO_INTSTATUS)

Stores the interrupt status after masking for Port A bits configured as interrupt sources

MEM Offset (B0000C00)	0B0000C40h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RO	26'b0	**Interrupt Status (GPIO_INTSTATUS)** After mask. See GPIO_RAW_INTSTATUS for raw interrupt values and GPIO_INTMASK for interrupt mask configuration

17.3.1.9 Raw Interrupt Status (GPIO_RAW_INTSTATUS)

MEM Offset (B0000C00)	0B0000C44h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RO	26'b0	**Raw Interrupt Status (GPIO_RAW_INTSTATUS)** Raw interrupt of status of Port A (premasking bits)

17.3.1.10 Debounce Enable (GPIO_DEBOUNCE)

Controls the debounce logic associated to a Port A bit configured as interrupt source

MEM Offset (B0000C00)	0B0000C48h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Debounce Enable (GPIO_DEBOUNCE)** Controls whether an external signal that is the source of an interrupt needs to be debounced to remove any spurious glitches. Writing a 1 to a bit in this register enables the debouncing circuitry. A signal must be valid for two periods of an external clock before it is internally processed. 0 No debounce (default) 1 Enable debounce

17.3.1.11 Clear Interrupt (GPIO_PORTA_EOI)

Controls edge-type interrupt clearing

MEM Offset (B0000C00)	0B0000C4Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RO	26'b0	**Clear Interrupt (GPIO_PORTA_EOI)** Controls the clearing of edge type interrupts from Port A. When a 1 is written into a corresponding bit of this register, the interrupt is cleared. All interrupts are cleared when Port A is not configured for interrupts. 0 No interrupt clear (default) 1 Clear interrupt

17.3.1.12 Port A External Port (GPIO_EXT_PORTA)

Used by the software to read values from the GPIO Port bits

MEM Offset (B0000C00)	0B0000C50h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RO	26'b0	**External Port (GPIO_EXT_PORTA)** Reading this location reads the values on the external signal.

17.3.1.13 Synchronization Level (GPIO_LS_SYNC)

Controls if a level-sensitive interrupt type need to be synchronized to the system clock

MEM Offset (B0000C00)	0B0000C60h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'b0	**Reserved (RSV)**
0	RW	1'b0	**Synchronization Level (GPIO_LS_SYNC)** Writing a 1 to this register results in all level-sensitive interrupts being synchronized to the system clock. 0 Not Synchronized (default) 1 Synchronized

17.3.1.14 Interrupt both edge type (GPIO_INT_BOTHEDGE)

Controls the edge type of interrupt that can occur on Port A

MEM Offset (B0000C00)	0B0000C68h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'b0	**Reserved (RSV0)**
24:0	RW	26'b0	**Interrupt both edge type (GPIO_PWIDTH_A)** Controls the edge type of interrupt that can occur on Port A. Whenever a particular bit is programmed to 1, it enables the generation of interrupts on both the rising edge and the falling edge of an external input signal corresponding to that bit on port A. The values programmed in the registers gpio_intype_level and gpio_int_polarity for this particular bit are not considered when the corresponding bit of this register is set to 1. Whenever a particular bit is programmed to 0, the interrupt type depends on the value of the corresponding bits in the gpio_inttype_level and gpio_int_polarity registers. 0 - Active-low (default) 1 - Active-high

17.3.1.15 GPIO Configuration Register 2 (GPIO_CONFIG_REG2)

Stores the bit Port width minus one

Register Offset	0B0000C70h
IntelRsvd	True
Size	32 bits
Default	0003_9C73h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	0003_9C73h	**RSVD (RSVD)** Reserved

17.3.1.16 GPIO Configuration Register 1 (GPIO_CONFIG_REG1)

Stores information on the GPIO controller configuration

Register Offset	0B0000C74h
IntelRsvd	True
Size	32 bits
Default	001F_70F6h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	001F_70F6h	**RSVD (RSVD)** Reserved

§

18 Timers and PWM

The Timer and Pulse Width Modulation (PWM) block allows individual control of the frequency and duty cycle of two output signals. The PWM block also supports use as a Timer block for the purposes of generating periodic interrupts.

18.1 Signal Descriptions

Please see Chapter 2, “Physical Interfaces” for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, “Electrical Characteristics”
- **Description:** A brief explanation of the signal’s function

Table 48. External Interface Signals

Signal Name	Direction/ Type	Description
PWM[0]	O / IOVDD	**Output 0:** *PWM Output 0*
PWM[1]	O / IOVDD	**Output 1:** *PWM Output 1*

18.2 Features

The following is a list of the PWM features:

- 2 Counters capable of operating in PWM Mode or Timer Mode
- PWM Mode
  - Configurable High and Low times for each PWM Output
  - High and Low time granularity of a single 32MHz clock period
  - Interrupt generation always on both the rising and falling edges of the PWM Output
  - Interrupt Control per PWM Output:
    - Interrupt Generation only on both edges of the PWM Output
    - Interrupt Mask Capability
- Timer Mode
  - 32-bit Timer operating at 32MHz
    - Timer Periods from 1 32MHz clock period (31.25ns) to 2^32-1 32MHz clock periods (134s)
  - Interrupt Control per Timer:
    - Interrupt Generation on Timer Expiry
    - Interrupt Mask Capability

18.2.1 PMW Signaling

The Timer and PWM block supports the generation of PWM Output signals with configurable low and high times which allows both the duty cycle and frequency to be set.

Some example PWM Output signals are shown in the following figures.

Figure 18. Duty Cycle of 20%

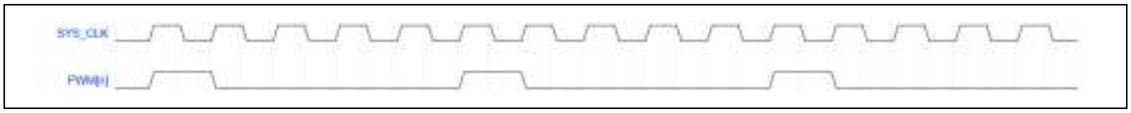


Figure 19. Duty Cycle of 50%

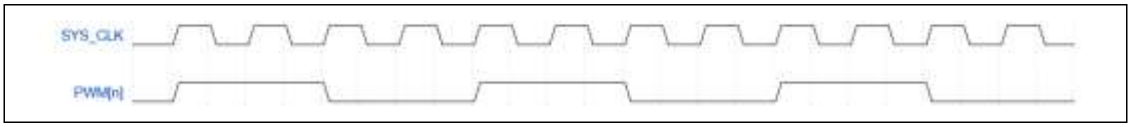
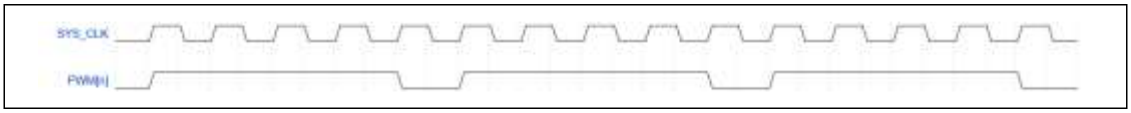


Figure 20. Duty Cycle of 80%



See Section 18.3.1 for details on configuring the low and high times.

18.2.2 Functional Operation

Each counter is identical, has an associated PWM Output and can be individually configured with the following options:

- Enable
- PWM Mode or Timer Mode
- PWM Duty Cycle and Frequency
- Timer Timeout Period
- Interrupt Masking

In PMW Mode the high and low times can be configured as follows. This assumes a nominal system clock frequency of 32MHz, the values, in nanoseconds, will differ if the system clock frequency is changed.

Table 49. PWM Timing

Characteristic	Value (System Clock Cycles)	Value (time)
Low Time Granularity	1	31.25ns
Low Time Range	2 to 4294967296 (2^32)	62.5ns to 134.22s
High Time Granularity	1	31.25ns

Characteristic	Value (System Clock Cycles)	Value (time)
High Time Range	2 to 4294967296 (2^32)	62.5ns to 134.22s

See Section 18.3.1 for details on configuring the low and high times.

PWM Mode supports the following maskable interrupt source:

- Both edges of the PWM Output signal.

In Timer Mode the timeout period can be configured as follows. This assumes a nominal system clock frequency of 32MHz, the values, in nanoseconds, will differ if the system clock frequency is changed.

Table 50. Timer Period

Characteristic	Value (System Clock Cycles)	Value (time)
Timeout Period Granularity	1	31.25ns
Timeout Period Range	0 to 4294967295 (2^32 -1)	0 to 134.22s

See Section 18.3.2 for details on configuring the timeout period.

Timer Mode supports the following maskable interrupt source:

- Timer Expiry.

Interrupts are cleared by reading the Timer N End Of Interrupt register.

18.3 Use

18.3.1 PWM Mode

Once enabled, the counter runs in free running mode and the associated PWM Output is set to 0.

The Low Time is determined by the Timer N Load Count register value which is loaded into counter upon enable, once the counter decrements to 0 the associated PWM Output toggles from 0 to 1 and the counter is loaded with the Timer N Load Count 2 register value which determines the High time.

When the counter subsequently decrements to 0 the associated PWM Output toggles from 1 to 0 and the timer is re-loaded with the Timer N Load Count register value and the process repeats.

18.3.2 Timer Mode

When a timer counter is enabled after being reset or disabled, the count value is loaded from the TimerNLoadCount register; this occurs in both free-running and user-defined count modes.

When a timer counts down to 0, it loads one of two values, depending on the timer operating mode:

- User-defined count mode – Timer loads the current value of TimerNLoadCount or TimerNLoadCount2 register alternatingly. One can program same value into TimerNLoadCount and

TimerNLoadCount2 registers for same interrupt periodicity. Use this mode if a fixed periodic timed interrupt is needed.

- Free-running mode – Timer loads the maximum value of 32-bits of all-ones. The timer counter wrapping to its maximum value allows time to reprogram or disable the timer before another interrupt occurs. Use this mode if a single timed interrupt is needed. After getting the interrupt, disable the timer so that it is stopped before it fires another interrupt after 2^{32} timer clock ticks.

In both the timer operating modes, timer counter is always running/decrementing at every timer clock tick, unless timer is disabled. An interrupt is generated when the timer count changes from 0 to its maximum count value.

In Timer Mode the PWM Outputs are unused and associated pins can be freed up for alternative functions by reconfiguring the pin muxing.

18.4 Memory Mapped IO Registers

Registers listed are for Timers, starting at base address B0000800h.

Table 51. Summary of PWM Registers—0xB0000800

MEM Address	Default	Instance Name	Name
0xB0000800	0000_0000h	Timer1LoadCount	Timer 1 Load Count
0xB0000804	0000_0000h	Timer1CurrentValue	Timer 1 Current Value
0xB0000808	0000_0000h	Timer1ControlReg	Timer 1 Control
0xB000080C	0000_0000h	Timer1EOI	Timer 1 End Of Interrupt
0xB0000810	0000_0000h	Timer1IntStatus	Timer 1 Interrupt Status
0xB0000814	0000_0000h	Timer2LoadCount	Timer 2 Load Count
0xB0000818	0000_0000h	Timer2CurrentValue	Timer 2 Current Value
0xB000081C	0000_0000h	Timer2ControlReg	Timer 2 Control
0xB0000820	0000_0000h	Timer2EOI	Timer 2 End Of Interrupt
0xB0000824	0000_0000h	Timer2IntStatus	Timer 2 Interrupt Status
0xB00008A0	0000_0000h	TimersIntStatus	Timers Interrupt Status
0xB00008A4	0000_0000h	TimersEOI	Timers End Of Interrupt
0xB00008A8	0000_0000h	TimersRawIntStatus	Timers Raw (unmasked) Interrupt Status
0xB00008AC	3230_392Ah	TimersCompVersion	Timers Component Version
0xB00008B0	0000_0000h	Timer1LoadCount2	Timer 1 Load Count 2
0xB00008B4	0000_0000h	Timer2LoadCount2	Timer 2 Load Count 2

18.4.1.1 Timer 1 Load Count (Timer1LoadCount)

MEM Offset (B0000800)	0B0000800h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Timer Load Count (TimerLoadCount)** In PWM mode, this register field controls the low period of pwm output pwm[0]. In timer mode, this is the value that the timer counter starts counting down from. Width of PWM LOW period = (TimerLoadCount + 1) * timer_N_clk clock period. Timer_N_clk is the peripheral clock connected to timer/pwm block.

18.4.1.2 Timer 1 Current Value (Timer1CurrentValue)

MEM Offset (B0000800)	0B0000804h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO	32'h0	**Timer Current Value (TimerCurrentValue)** In both PWM and Timer mode, reading this register field returns the current value of the counter controlling the PWM/Timer. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.3 Timer 1 Control (Timer1ControlReg)

MEM Offset (B0000800)	0B0000808h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'h0	**Reserved (RSVD1)** Reserved

Bits	Access Type	Default	Description
3	RW	1'h0	**Timer PWM (TIMER_PWM)** Select between PWM mode and Timer mode. 1 - PWM Mode 0 - Timer Mode
2	RW	1'h0	**Timer Interrupt Mask (TIMER_INTERRUPT_MASK)** Set to b1 to mask the interrupt from PWM/Timer
1	RW	1'h0	**Timer Mode (TIMER_MODE)** Select between free running and user defined count mode. 1 - user-defined count mode 0 - free-running mode NOTE : PWM timer must be operated in user-defined count mode i.e. this field must be set to b1
0	RW	1'h0	**Timer Enable (TIMER_ENABLE)** 0 - Disable PWM/Timer 1 - Enable PWM/Timer

18.4.1.4 Timer 1 End Of Interrupt (Timer1EOI)

MEM Offset (B0000800)	0B000080Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Timer End-of-Interrupt (TIMER_END_OF_INTERRUPT)** Reading from this register returns b0, and clears the interrupt form PWM/Timer.

18.4.1.5 Timer 1 Interrupt Status (Timer1IntStatus)

MEM Offset (B0000800)	0B0000810h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Timer Interrupt Status (TIMER_INTERRUPT_STATUS)** A read of this register returns the post masking interrupt status of PWM/Timer. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.6 Timer 2 Load Count (Timer2LoadCount)

MEM Offset (B0000800)	0B0000814h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Timer Load Count (TimerLoadCount)** In PWM mode, this register field controls the low period of pwm_o. In timer mode, this is the value that the timer counter starts counting down from, and controls the high and low periods of pwm_o.

18.4.1.7 Timer 2 Current Value (Timer2CurrentValue)

MEM Offset (B0000800)	0B0000818h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO	32'h0	**Timer Current Value (TimerCurrentValue)** In both PWM and Timer mode, reading this register field returns the current value of the counter controlling the PWM/Timer. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.8 Timer 2 Control (Timer2ControlReg)

MEM Offset (B0000800)	0B000081Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'h0	**Reserved (RSVD1)** Reserved
3	RW	1'h0	**Timer PWM (TIMER_PWM)** Select between PWM mode and Timer mode. 1 - PWM Mode 0 - Timer Mode
2	RW	1'h0	**Timer Interrupt Mask (TIMER_INTERRUPT_MASK)** Set to b1 to mask the interrupt from PWM/Timer
1	RW	1'h0	**Timer Mode (TIMER_MODE)** Select between free running and user defined count mode. 1 - user-defined count mode 0 - free-running mode NOTE : PWM timer must be operated in user-defined count mode i.e. this field must be set to b1
0	RW	1'h0	**Timer Enable (TIMER_ENABLE)** 0 - Disable PWM/Timer 1 - Enable PWM/Timer

18.4.1.9 Timer 2 End Of Interrupt (Timer2EOI)

MEM Offset (B0000800)	0B0000820h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Timer End-of-Interrupt (TIMER_END_OF_INTERRUPT)** Reading from this register returns b0, and clears the interrupt form PWM/Timer.

18.4.1.10 Timer 2 Interrupt Status (Timer2IntStatus)

MEM Offset (B0000800)	0B0000824h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Timer Interrupt Status (TIMER_INTERRUPT_STATUS)** A read of this register returns the post masking interrupt status of PWM/Timer. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.11 Timers Interrupt Status (TimersIntStatus)

MEM Offset (B0000800)	0B00008A0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h0	**Reserved (RSVD1)** Reserved
7:0	RO	8'h0	**Timers Interrupt Status (TIMERS_INTERRUPT_STATUS)** A read of this register returns the post masking interrupt status of PWM/Timer's 0 to 7. Bit corresponds to PWM/Timer. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.12 Timers End Of Interrupt (TimersEOI)

MEM Offset (B0000800)	0B00008A4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h0	**Reserved (RSVD1)** Reserved
7:0	RO	8'h0	**Timers End-of-Interrupt Status (TIMERS_END_OF_INTERRUPT)** A read of this register returns all 0's, and clears all active interrupts from all PWM/Timers.

18.4.1.13 Timers Raw (unmasked) Interrupt Status (TimersRawIntStatus)

MEM Offset (B0000800)	0B00008A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h0	**Reserved (RSVD1)** Reserved
7:0	RO	8'h0	**Timers Raw Interrupt Status (TIMERS_RAW_INTERRUPT_STATUS)** A read of this register returns the pre masking interrupt status of all PWM/Timers. If reading from this register outside of an interrupt service routine, software should first perform 2 dummy writes to another PWM/Timer register.

18.4.1.14 Timers Component Version (TimersCompVersion)

MEM Offset (B0000800)	0B00008ACh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	3230_392Ah

Bits	Access Type	Default	Description
31:0	RO	32'h3230392a	**Timers Component Version (TIMERS_COMPONENT_VERSION)** A read of this register returns an ASCII string representing the version number of the Timers Logic.

18.4.1.15 Timer 1 Load Count 2 (Timer1LoadCount2)

MEM Offset (B0000800)	0B00008B0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Timer Load Count 2 (TimerLoadCount2)** In PWM mode, this register field controls the high period of pwm_o. In timer mode, this register has no functional use.

18.4.1.16 Timer 2 Load Count 2 (Timer2LoadCount2)

MEM Offset (B0000800)	0B00008B4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Timer Load Count 2 (TimerLoadCount2)** In PWM mode, this register field controls the high period of pwm_o. In timer mode, this register has no functional use.

§

19 Watchdog Timer

The Watchdog Timer can be used to trigger a Warm Reset in the event that the SoC has become unresponsive.

19.1 Features

The following is a list of the Watchdog (WDT) features:

- Timer can be disabled (default state) or locked enabled (SoC Warm Reset required to disable)
- Selectable Timeout Value that ranges from 8us to ~60s (32MHz)
- Capability to have a different initial Timeout Value versus the reload Timeout Value
- 2 Timeout Response Modes as Follows:
  - Request a SoC Warm Reset when a timeout occurs.
  - Generate an Interrupt when a timeout occurs and if the Interrupt is not serviced by the time a second timeout occurs then requests a SoC Warm Reset.

19.1.1 WDT Enable

The WDT_CR.WDT_EN register field must be set to 1 to enable the WDT, once set to 1 the WDT_EN field can only be set back to 0 by an SoC Reset.

19.1.2 WDT Timeout Capabilities

The timer uses a 32-bit down-counter which is loaded with the programmed Timeout Value.

The Initial Timeout Value is selected by the WDT_TORR.TOP_INIT register field (which is fixed to 0), this value gets loaded into the timer when the WDT is first enabled. The Reload Timeout Value is selected by the WDT_TORR.TOP register field, this value gets loaded into the timer on subsequent reloads of the timer.

The values below are based off a 32 MHz System Clock and must be adjusted if the frequency is adjusted (see Clocking Section).

Table 52. WDT Timeout Selection

TOP_INIT / TOP	Clock Cycles	Value (at 32MHz)
0	2^{16}	2.048ms
1	2^{17}	4.096ms
2	2^{18}	8.192ms
3	2^{19}	16.384ms
4	2^{20}	32.768ms
5	2^{21}	65.536ms

6	2^{22}	131.072ms
7	2^{23}	262.144ms
8	2^{24}	524.288ms
9	2^{25}	1.049s
10	2^{26}	2.097s
11	2^{27}	4.194s
12	2^{28}	8.389s
13	2^{29}	16.777s
14	2^{30}	33.554s
15	2^{31}	67.109s

19.2 Use

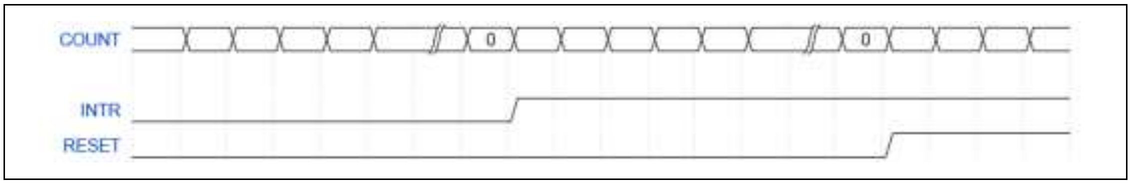
When enabled the timer starts counting down from the programmed Timeout Value. If the processor fails to reload the counter before it reaches zero (timeout) the WDT will do one of two things depending on the programmed Response Mode:

Table 53. WDT Response Mode

Response Mode	Behavior
0	The WDT requests a SoC Warm Reset on a timeout.
1	The WDT generates an interrupt on first timeout. If interrupt is not cleared by the second timeout, the WDT requests a SoC warm reset. See Figure 54. WDT Behavior for Response Mode of 1.

NOTE: When the counter reaches zero, it wraps to the programmed Timeout Value and continues decrementing.

Figure 21. WDT Behaviour for Response Mode of 1



The counter is reloaded by writing 76h to the Counter Restart Register, this will also clear the WDT Interrupt.

The WDT Interrupt may also be cleared by reading the Interrupt Clear Register, however this will not reload the counter.

19.3 Memory Mapped IO Registers

Registers listed are for the WDT, starting at base address B0000000h

Table 54. Summary of WDT Registers—0xB0000000

MEM Address	Default	Instance Name	Name
0x0	0000_0002h	WDT_CR	Control Register
0x4	0000_0000h	WDT_TORR	Timeout Range Register
0x8	0000_FFFFh	WDT_CCVR	Current Counter Value Register
0xC	0000_0000h	WDT_CRR	Current Restart Register
0x10	0000_0000h	WDT_STAT	Interrupt Status Register
0x14	0000_0000h	WDT_EOI	Interrupt Clear Register
0xE4	0000_0000h	WDT_COMP_PARAM_5	Component Parameters
0xE8	0000_0000h	WDT_COMP_PARAM_4	Component Parameters
0xEC	0000_0000h	WDT_COMP_PARAM_3	Component Parameters
0xF0	0000_0000h	WDT_COMP_PARAM_2	Component Parameters
0xF4	1000_0242h	WDT_COMP_PARAM_1	Component Parameters Register 1
0xF8	3130_372Ah	WDT_COMP_VERSION	Component Version Register
0xFC	4457_0120h	WDT_COMP_TYPE	Component Type Register

19.3.1.1 Control Register (WDT_CR)

MEM Offset (B0000000)	0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0002h

Bits	Access Type	Default	Description
31:6	RO	26'h0	**Reserved (RSVD1)** Reserved
5	RW	1'h0	**Reserved**

Bits	Access Type	Default	Description
4:2	RW	3'h0	**Reset Pulse Width (RST_PULSE_WIDTH)** Rest pulse length 000 - 2 pclk cycles 001 - 4 pclk cycles 010 - 8 pclk cycles 011 - 16 pclk cycles 100 - 32 pclk cycles 101 - 64 pclk cycles 110 - 128 pclk cycles 111 - 256 pclk cycles
1	RW	1'h1	**RMOD (RMOD)** Response mode 0 = Generate a system reset 1 = First generate interrupt if not cleared generate reset
0	RW	1'h0	**WDT Enable (WDT_ENABLE)** WDT Enable 0 = WDT Disable 1 = WDT Enable

19.3.1.2 Timeout Range Register (WDT_TORR)

MEM Offset (B0000000)	4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h0	**Reserved (RSVD1)** Reserved
7:4	RW	4'h0	**Timeout period for initialization (TOP_INIT)** Writes to these register bits have no effect when the configuration parameter WDT_HC_TOP = 1 or WDT_ALWAYS_EN = 1. Used to select the timeout period that the watchdog counter restarts from for the first counter restart (kick). This register should be written after reset and before the WDT is enabled.
3:0	RW	4'h0	**Timeout period (TOP)** Writes have no effect when the configuration parameter WDT_HC_TOP = 1, thus making this register read-only. This field is used to select the timeout period from which the watchdog counter restarts. A change of the timeout period takes effect only after the next counter restart (kick).

19.3.1.3 Current Counter Value Register (WDT_CCVR)

MEM Offset (B0000000)	8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_FFFFh

Bits	Access Type	Default	Description
31:0	RW	32'hFFFF	**Current Counter Value Register (WDT_CCVR)** This register when read is the current value of the internal counter.

19.3.1.4 Current Restart Register (WDT_CRR)

MEM Offset (B0000000)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Current Restart Register (WDT_CRR)** This register is used to restart the WDT counter. The value 0x76 must be written as a safety feature to prevent accidental restarts. A restart also clears the WDT interrupt. Reading this register returns zero.

19.3.1.5 Interrupt Status Register (WDT_STAT)

MEM Offset (B0000000)	10h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Interrupt Status Register (WDT_STAT)** Theis register shows the interrupt status of the WDT. 0 = interrupt is active 1 = interrupt is inactive

19.3.1.6 Interrupt Clear Register (WDT_EOI)

MEM Offset (B0000000)	14h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**Interrupt Clear Register (WDT_EOI)** Clears the watchdog interrupt. This can be used to clear the interrupt without restarting the watchdog counter.

19.3.1.7 Component Parameters (WDT_COMP_PARAM_5)

Register Offset	0E4h
IntelRsvd	True
Size	32 bits
Default	0000_0000h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	0000_0000h	**RSVD (RSVD)** Reserved

19.3.1.8 Component Parameters (WDT_COMP_PARAM_4)

Register Offset	0E8h
IntelRsvd	True
Size	32 bits
Default	0000_0000h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	0000_0000h	**RSVD (RSVD)** Reserved

19.3.1.9 Component Parameters (WDT_COMP_PARAM_3)

Register Offset	0ECh
IntelRsvd	True
Size	32 bits
Default	0000_0000h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	0000_0000h	**RSVD (RSVD)** Reserved

19.3.1.10 Component Parameters (WDT_COMP_PARAM_2)

Register Offset	0F0h
IntelRsvd	True
Size	32 bits
Default	0000_0000h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	0000_0000h	**RSVD (RSVD)** Reserved

19.3.1.11 Component Parameters Register 1 (WDT_COMP_PARAM_1)

Register Offset	0F4h
IntelRsvd	True
Size	32 bits
Default	1000_0242h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	1000_0242h	**RSVD (RSVD)** Reserved

19.3.1.12 Component Version Register (WDT_COMP_VERSION)

Register Offset	0F8h
IntelRsvd	True
Size	32 bits
Default	3130_372Ah
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	3130_372Ah	**RSVD (RSVD)** Reserved

19.3.1.13 Component Type Register (WDT_COMP_TYPE)

Register Offset	0FCh
IntelRsvd	True
Size	32 bits
Default	4457_0120h
PowerWell	

Bits	Access Type	Default	Description
31:0	RO	4457_0120h	**RSVD (RSVD)** Reserved

§

20 Real Time Clock (RTC)

The SoC contains a Real Time Clock for the purpose of keeping track of time. The RTC operates from 1 Hz to 32.768 kHz.

The RTC supports alarm functionality that allows scheduling an Interrupt / Wake Event for a future time.

The RTC operates in all SoC Power States. The RTC is powered from the same battery supply as the rest of the SOC and does not have its own dedicated supply.

20.1 Signal Descriptions

Please see Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, "Electrical Characteristics"
- Description: A brief explanation of the signal's function

Table 55. Memory Signals

Signal Name	Direction/ Type	Description
OSC32_IN	I Analog	**Crystal Input:** This signal is connected to the 32.768 kHz Crystal.
OSC32_OUT	O Analog	**Crystal Output:** This signal is connected to the 32.768 kHz Crystal.

20.2 Features

The following is a list of the RTC features:

- Programmable 32-bit binary Counter
- Counter increments on successive edges of a Counter Clock from 1 Hz to 32.768 kHz (derived from the 32.768 kHz Crystal Oscillator clock)
- Comparator for Interrupt / Wake Event generation based on the programmed Match Value
- Supports Interrupt / Wake Event generation when only the Counter Clock is running (Fabric Clock is off)

20.2.1 RTC Clock

The RTC clock is the output of a 4bit prescaler (see Clocking Section) that is driven by the output of the 32.768 kHz Crystal Oscillator.

This allows a range of clock frequencies to be generated as shown in the table below:

Table 56. RTC Clock Scaling

Scaling Factor	Output Clock Frequency
0	32.768 kHz
1	16.384 kHz
3	8.192 kHz
4	4.096 kHz
5	2.048 kHz
6	1.024 kHz
7	512 Hz
8	256 Hz
9	128 Hz
10	64 Hz
11	32 Hz
12	16 Hz
13	8 Hz
14	4 Hz
15	2 Hz

20.2.2 Counter Functionality

The RTC contains a single 32bit up-counter, the counter starts to increment when the RTC is taken out of reset. The counter increments on successive edges of the RTC clock.

The counter Start Value is loaded by writing a 32bit value to the Counter Load Register. The Counter supports loading a Start Value while it is incrementing.

When the counter reaches its max value (2^{32}-1) it wraps to 0 and continues incrementing.

The RTC supports reading the Counter via the read-only Counter Value Register.

For accessing any registers in RTC block, both APN Clock (pclk) and counter clock (rtcclk) should be running and the frequency of the APB Bus clock must be greater than three times the frequency of the counter clock. If CCU_SYS_CLK_SEL register in SCSS is configured to choose rtcclk for system clock, then CCU_RTC_CLK_DIB register in SCSS shall be configured for div-by-4 or lower, in order to ensure that all register accesses to RTC block are properly synchronized to counter clock (rtcclk) domain.

After writing a register, firmware has to wait for at least 1 rtcclk input before clock gating pclk. Interrupt status bit (rtc_stat) in RTC_STAT/RTC_RSTAT register will get cleared 1 rtcclk period after reading RTC_EOI register. Alternatively firmware can choose to ignore the interrupt status register and rely only on the rtc_intr line interrupt for interrupt status, which gets cleared immediately after reading RTC_EOI register.

20.3 Use

The RTC allows the user to disable interrupt generation and also to mask a generated interrupt.

20.3.1 Clock and Calendar

The 32bit counter is intended to provide Clock and Calendar functionality, the capabilities differ depending on the chosen frequency of the RTC clock as described in the examples below.

Using a 1 Hz RTC clock the following capabilities are exposed:

- Counter increments every second
- Counter wraps in 2^{32}-1 seconds (~136 years)
- Counter can be used to store Time and Date in the Unix Time format defined as the number of seconds since 00:00:00 UTC on 1 January 1970 (the epoch)

Using a 32.68 kHz RTC clock the following capabilities are exposed:

- Counter increments every 30.5 microsecond
- Counter wraps in 2^{32}-1 * 30.5 microsecond (~1.51 days)

20.3.2 Alarm

Alarm functionality is provided by the Match Value Register, the interrupt generation logic asserts the interrupt (if enabled) when the Counter reaches this Match Value.

The RTC allows the user to disable interrupt generation and also to mask a generated interrupt.

Additionally the RTC supports generation of an interrupt when only the RTC clock is running, this allows the interrupt to be generated when in the Deep Sleep state.

20.3.3 Wake Event

The RTC supports waking the SoC from Low Power States, including Sleep.

The SoC use the RTC interrupt as the source of this wake event so interrupt generation must be enabled to facilitate this RTC wake capability.

20.4 Memory Mapped IO Registers

Registers listed are for RTC, starting at base address B000400h.

Table 57. Summary of RTC Registers—0xB0000400

MEM Address	Default	Instance Name	Name
0xB0000400	0000_0000h	RTC_CCVR	Current Counter Value Register
0xB0000404	0000_0000h	RTC_CMR	Current Match Register

MEM Address	Default	Instance Name	Name
0xB0000408	0000_0000h	RTC_CLR	Counter Load Register
0xB000040C	0000_0000h	RTC_CCR	Counter Control Register
0xB0000410	0000_0000h	RTC_STAT	Interrupt Status Register
0xB0000414	0000_0000h	RTC_RSTAT	Interrupt Raw Status Register
0xB0000418	0000_0000h	RTC_EOI	End of Interrupt Register
0xB000041C	3230_332Ah	RTC_COMP_VERSION	Component Version

20.4.1.1 Current Counter Value Register (RTC_CCVR)

MEM Offset (B0000400)	0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO	32'h0	**Current Counter Value Register (RTC_CCVR)** This register when read is the current value of the internal counter.

20.4.1.2 Current Match Register (RTC_CMR)

MEM Offset (B0000400)	4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Current Match Register (RTC_CMR)** When the internal counter matches this register and interrupt is generated, provided interrupt generation is enabled.

20.4.1.3 Counter Load Register (RTC_CLR)

MEM Offset (B0000400)	8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**Counter Load Register (RTC_CLR)** Loaded into the counter as the loaded value which is written coherently.

20.4.1.4 Counter Control Register (RTC_CCR)

MEM Offset (B0000400)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	PowerWell
32.768 kHz	32.768 kHz	32.768 kHz	32.768 kHz	32.768 kHz
16.384 kHz	16.384 kHz	16.384 kHz	16.384 kHz	16.384 kHz
8.192 kHz	8.192 kHz	8.192 kHz	8.192 kHz	8.192 kHz
4.096 kHz	4.096 kHz	4.096 kHz	4.096 kHz	4.096 kHz
2.048 kHz	2.048 kHz	2.048 kHz	2.048 kHz	2.048 kHz

20.4.1.5 Interrupt Status Register (RTC_STAT)

MEM Offset (B0000400)	10h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**RTC STAT (RTC_STAT)** This register is the interrupt status. 0 = interrupt is inactive 1 = interrupt is active

20.4.1.6 Interrupt Raw Status Register (RTC_RSTAT)

MEM Offset (B0000400)	14h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**RTC RSTAT (RTC_RSTAT)** This register is the raw interrupt status. 0 = interrupt is inactive 1 = interrupt is active

20.4.1.7 End of Interrupt Register (RTC_EOI)

MEM Offset (B0000400)	18h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSVD1)** Reserved
0	RO	1'h0	**RTC EOI (RTC_EOI)** By reading this location the match interrupt is cleared.

20.4.1.8 End of Interrupt Register (RTC_COMP_VERSION)

Register Offset	1Ch
IntelRsvd	True
Size	32 bits
Default	0000_0000h
PowerWell	

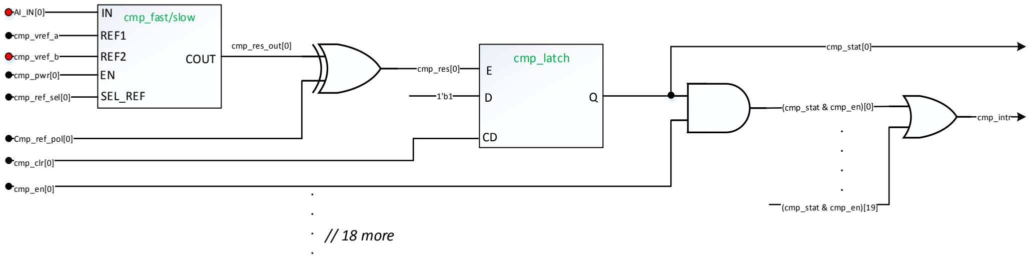
Bits	Access Type	Default	Description
31:0	RO	0000_0000h	**RSVD (RSVD)** Reserved

§

21 Comparators

The SOC supports 19 low power comparators which can be used to wake the system from low power states. Two types of comparators are supported, a low power, low performance version and a high power high performance version. Analog Inputs [5:0] are connected to high performance comparators and Analog Inputs [18:6] are connected to low power comparators.

Each comparator can be powered down to achieve even lower power. Comparator reference supply is selectable between the internal VREF (0.95V +/- 10%) and an external user supplied reference (AR input).



21.1 Signal Descriptions

Table 58. Memory Signals

Signal Name	Direction/ Type	Description
AI[18:0]	I	**External Comparator input**
AR	I	**External Comparator reference voltage**

21.2 Features

The following is a list of comparator features:

- 2.0V – 3.63V AVDD operation
- 1.2V – 1.98V DVDD operation
- Fast Asynchronous comparator
- Rail-to-rail input range
- CMPLP
  - <3.8us propagation delay
  - <600nA static current
  - <10mV hysteresis (6 mV typ)
  - <22nA power down current
- CMPHP

- <0.25us propagation delay
- <9.8uA static current
- <6.8mV hysteresis (4.6 mV typ)
- <2.7nA power down current

21.3 Use

The 19 comparators can be used in the following ways:

- To generate an interrupt to the processor
- To generate a wake event to cause the PMU to exit a deep sleep condition.

The following sequence should be applied setting up the comparator control to generate an interrupt or wake event.

1. Set CMP_REF_SEL for each comparator
2. Set CMP_REF_POL for each comparator
3. Set CMP_PWR for each comparator to ‘1’
4. Set CMP_EN for each comparator to ‘1’

The comparator interrupt is a level triggered interrupt based on the polarity set in CMP_REF_POL. It is not edge triggered. Interrupt is latched when external analog input matches that of CMP_REF_POL and this latch is cleared using CMP_STAT_CLR. Interrupt persists as long as the external source maintains its signal state as that of CMP_REF_POL, and the particular comparator is enabled (CMP_EN[x]=1).

The following sequence should be used when responding to an interrupt having being asserted.

1. Read CMP_STAT_CLR for each comparator
2. Clear CMP_STAT_CLR for each firing comparator by writing a ‘1’

If the external analog input generates a pulse matching polarity level of CMP_REF_POL for sufficient duration greater than comparator’s propagation delay, interrupt gets latched. If external input maintains its signal state to that of CMP_REF_POL, interrupt persists and is not cleared even if CMP_STAT_CLR is applied.

The comparators are directly connected to the PMU logic in the always on domain of the SOC. When a comparator activates the PMU will exit its low power wait or deep sleep states. (See Chapter 8, “Power Management” for more details)

§

22 Analog to Digital Convertor (ADC)

The SoC implements a Successive-Approximation (SAR) Analog to Digital Convertor (ADC) which is capable of taking 19 single-ended analog inputs for conversion. ADC is characterized to operate over the AVDD (1.8 to 3.6 V) analog input range.

22.1 Signal Descriptions

Please see Chapter 2, "Physical Interfaces" for additional details.

The signal description table has the following headings:

- **Signal Name:** The name of the signal/pin
- **Direction:** The buffer direction can be either input, output, or I/O (bidirectional)
- **Type:** The buffer type found in Chapter 4, "Electrical Characteristics"
- **Description:** A brief explanation of the signal's function

Table 59. Memory Signals

Signal Name	Direction/ Type	Description
ADC[18:0]	I	Comparator/ADC inputs

22.2 Features

The following is a list of the ADC features:

- 19:1 multiplexed single-ended analog input channels, 6 high speed inputs and 13 low speed inputs.
- Selectable Resolution between 12, 10, 8 and 6-bit (12-bit at 2.28 MSps and 6 bits at 4 MSps).
  - Max achievable sampling rate = (adc clock frequency) / (selres + 2).
- ADC Parameters:
  - Differential Non-Linearity DNL = +/- 1.0 LSB
  - Integral Non-Linearity INL = +/- 2.0 LSB
  - SINAD = 68 dBFS
  - Offset Error = +/- 2 LSB (calibration enabled), +/- 64 LSB (calibration disabled)
- Latencies:
  - Power-up time of <= 10 us
  - 1 Conversion cycle = (resolution bits + 2) cycles

- Full-scale input range is 0 to AVDD.
  - ADC Reference Voltage (Vrefp) of ADC HIP is connected to AVDD.

- Current Consumption:
  - ~18 uA at 10 kSPS
  - ~240 uA at 1 MSPS

- ~1.1 mA at 5 MSPS
- ~15 uA standby
- ~2 uA powerdown

Notes:

1. ADC Hard macro takes in adcclk in the range of 140 kHz to 32 MHz. Minimum clock frequency is 140 kHz. adcclk is derived from system clock by configuring CCU_PERIPH_CLK_DIV_CTL0.CCU_ADC_CLK_DIV register in SCSS.
2. ADC Hard macro at its max sampling rate takes selres resolution + 2 cycles. For 12-bit resolution, it takes 14 cycles to get a sample. To read the sample by processor/DMA, it takes 3 system clock cycles minimum. Depending on number of cycles spent by processor for interrupt servicing and to process a given set of samples, max achievable sampling rate at system level could be lesser than or equal to max possible sampling rate (12-bit at 2.28 MSps, 10-bit at 2.6 MSps, 8-bit at 3.2 MSps and 6 bits at 4 MSps).
3. Minimum sampling rate is a function of adclk's min frequency of 140 kHz = 140 kHz / (selres + 2). Additionally there is a sampling window SW[7:0] register to delay sampling. For any sampling rate below this limit, software has to time and then trigger conversion accordingly.

22.3 Use

After powerup/cold reset, ADC is in deep power down state. Depending on current consumption and entry/exit latencies involved, firmware can put the ADC in different power states. When ADC is not in use, it is better to put it in low power states to reduce current consumption.

Power State	Definition	Max Current	Entry Latency	Exit Latency	How Triggered
ON	Normal Operation. ADC is enabled for conversation with optionally enabled internal LDO. Enadc=H, enldo=H, dislvl=L. powerup time: 3-5-10 usec (min-typ-max).	220 uA @ avdd, 20 uA @ dvdd at 1 MSps	-	-	Writing to ADC_OP_MODE register
STBY	Standby Mode. ADC is disabled but ADC state is kept enabled by enabling internal LDO and retaining DVDD. Enadc=L, enldo=H, dislvl=L. Exit involves 1 conversion cycle.	15 uA @ avdd, 1 uA @ dvdd	-	14 CLK cycles	Writing to ADC_OP_MODE register
PD	Power Down mode. ADC is disabled, calibration state is retained, DVDD is present, internal LDO is off. Enadc=L, enldo=L, dislvl=L. A calibrated conversion cycle can start immediately after internal LDO Power Up time.	1 uA @ avdd, 1 uA @ dvdd	-	10 usec	Writing to ADC_OP_MODE register
DPD	Deep Power Down mode. ADC is disabled, calibration state is lost, DVDD can be off. Enadc=L, enldo=L, dislvl=H. exit involves waiting for internal voltage regulator to start-up + recalibration + dummy conversion cycle. A complete calibration cycle lasts 81 clock cycles. A conversion cycle is 14 CLK cycles for 12-bit resolution.	1 uA @ avdd, 0.5 uA @ dvdd	-	10 usec + 95 CLK cycles.	Writing to ADC_OP_MODE register

Whenever exit from deep power down mode or at power up/cold reset, Calibration has to be performed to reduce offset error from +/- 64 LSB to +/- 2 LSB.

At Power-up / cold reset cycle / transitioning from any other power modes (Deep Power Down or Power Down or Standby) to normal mode (with calibration or without calibration), below steps are to be performed:

1. Issue “Normal with/without calibration” command through ADC_OP_MODE register (OM=”Normal with calibration” or “Normal Without Calibration”, Delay = 500). Note that Calibration is required if offset error (+/- 64 LSB) has to be reduced to +/- 2 LSB.
2. Issue “Start Calibration” command by writing ADC_command = Start calibration in ADC Command Register. This step is optional if calibration is not required. Optionally “Load Calibration” command can be performed if previous calibration word is restored.
   (AND/OR)
   Issue a dummy conversion cycle. Dummy conversion cycle is not required if “Start calibration” is performed. This is done by:

   a. Setting ADC channel sequence table with one entry (channel = 0).
   b. Issue ADC Command, with sampling window SW=0, Number of samples NS = 0 (1 sample), ADC_command = Start Single Shot Conversion.
3. Use ADC as required.
4. Later on, depending on whether ADC is in use or not, power state can be lowered to standby/powerdown depending on required power saving and acceptable entry/exit latency. Before the soc/system is put into “Deep sleep” power state, ADC has to be put to “Deep Power Down” state.

For converting analog input to digital sample once ADC is in normal/ON power state with calibration completed, the programming sequence for doing either single shot conversion or continuous conversion is:

1. Setup ADC Interrupt Enable register.
2. Setup ADC Channel Sequence table based on channel inputs to be converted.
3. Issue ADC Command, with setting sampling window, resolution, ADC_command = Start Single Conversion or Start Continuous Conversion. Number of Samples is used to stop single conversion once configured total number of samples (NS+1) are collected from channels as per channel sequence table. In case of continuous conversion, interrupt is raised every Ns number of samples are collected.

A continuous conversion can later be stopped by issuing ADC command with ADC_command = Stop Continuous conversion.

22.4 Memory Mapped IO Registers

Registers listed are for ADC, starting at base address B0004000h.

Table 60. Summary of ADC Registers—0xB0004000

MEM Address	Default	Instance Name	Name
0x0	8080_8080h	ADC_SEQ[0]	ADC Channel Sequence Table

0x4	8080_8080h	ADC_SEQ[1]	ADC Channel Sequence Table
0x8	8080_8080h	ADC_SEQ[2]	ADC Channel Sequence Table
0xC	8080_8080h	ADC_SEQ[3]	ADC Channel Sequence Table
0x10	8080_8080h	ADC_SEQ[4]	ADC Channel Sequence Table
0x14	8080_8080h	ADC_SEQ[5]	ADC Channel Sequence Table
0x18	8080_8080h	ADC_SEQ[6]	ADC Channel Sequence Table
0x1C	8080_8080h	ADC_SEQ[7]	ADC Channel Sequence Table
0x20	0000_0000h	ADC_CMD	ADC Command Register
0x24	0000_0000h	ADC_INTR_STATUS	ADC Interrupt Status Register
0x28	0000_0000h	ADC_INTR_ENABLE	ADC Interrupt Enable
0x2C	0000_0000h	ADC_SAMPLE	ADC Sample Register
0x30	0000_0000h	ADC_CALIBRATION	ADC Calibraton Data Register
0x34	0000_0000h	ADC_FIFO_COUNT	ADC FIFO Count Register
0x38	0000_0FA0h	ADC_OP_MODE	ADC Operating Mode Register

22.4.1.1 ADC Channel Sequence Table (ADC_SEQ [0..7])

MEM Offset (00000000)	[0]:0h [1]:4h [2]:8h [3]:0Ch [4]:10h [5]:14h [6]:18h [7]:1Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	8080_8080h

Bits	Access Type	Default	Description
31	RW	1'h1	**Last[4N+3] (Last_3)** Last Channel when set, this bit indicates that this is the last channel in the sequence. Subsequent conversions will begin at the start of the table.
30:29	RO	2'h0	**RSVD (RSVD)** Reserved
28:24	RW	5'h0	**Channel[4N+3] (Channel_3)** ADC Channel: this bit field defines the ADC channel to sample
23	RW	1'h1	**Last[4N+2] (Last_2)** Last Channel when set, this bit indicates that this is the last channel in the sequence. Subsequent conversions will begin at the start of the table.
22:21	RO	2'h0	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
20:16	RW	5'h0	**Channel[4N+2] (Channel_2)** ADC Channel: this bit field defines the ADC channel to sample
15	RW	1'h1	**Last[4N+1] (Last_1)** Last Channel when set, this bit indicates that this is the last channel in the sequence. Subsequent conversions will begin at the start of the table.
14:13	RO	2'h0	**RSVD (RSVD)** Reserved
12:8	RW	5'h0	**Channel[4N+1] (Channel_1)** ADC Channel: this bit field defines the ADC channel to sample
7	RW	1'h1	**Last[4N+0] (Last_0)** Last Channel when set, this bit indicates that this is the last channel in the sequence. Subsequent conversions will begin at the start of the table.
6:5	RO	2'h0	**RSVD (RSVD)** Reserved
4:0	RW	5'h0	**Channel[4N+0] (Channel_0)** ADC Channel: this bit field defines the ADC channel to sample

22.4.1.2 ADC Command Register (ADC_CMD)

MEM Offset (00000000)	20h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

This register returns 0x0 value when ADC Calibration & Conversion is in IDLE state waiting for a new command. When a command is written and is not completed yet (state machine not in IDLE state), then reading back this register returns value written as part of issuance of command.

Bits	Access Type	Default	Description
31:24	RW	8'h0	**SW - Sample Window (SW)** When the ADC command is a conversion command, this bit field defines the length of the sample interval in ADC clocks. This field is encoded thusly: sampleInterval=SW+2, sampleRate=adcFreq/(N+sampleInterval)
23	RO	1'h0	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
22:16	RW	7'h0	**CDI - Calibration Data Input (CDI)** When the ADC command is a load calibration command, this bit field defines the digital offset calibration data to be loaded.
15:14	RW	2'h0	**Resolution (Resolution)** This bit field defines the number of bits of precision. It is defined thus: 0 - 6bits 1 - 8bits 2 - 10bits 3 - 12bits
13:9	RW	5'h0	**CSTI - Channel Sequence Table Index (CSTI)** When the ADC command is a start conversion command, this bit field defines the first entry in the channel sequence table to be sampled. If repeated or continuous conversions are commanded, subsequent conversions will follow the channel sequence table.
8:4	RW	5'h0	**NS - Number of Samples (NS)** When the ADC command is start single conversion, this bit field defines the number of repeated conversions before stopping. When the ADC command is start continuous conversions, this bit field defines the number of conversion between interrupts. This bit field is ignored otherwise. It is encoded thus: numConversions=NS+1
3	RW	1'h0	**IE - Interrupt Enable (IE)** When set, this bit enables an interrupt on completion of the ADC operation.
2:0	RW	3'h0	**ADC Command (ADC_Command)** This bit field defines ADC operation. It is encoded thus: 0 = Start single conversion 1 = Start continuous conversion 2 = Reset calibration 3 = Start calibration 4 = Load calibration 5 = Stop continuous conversion 6 = No operation 7 = No operation

22.4.1.3 ADC Interrupt Status Register (ADC_INTR_STATUS)

MEM Offset (00000000)	24h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'h0000_0000	**RSVD (RSVD)** Reserved
2	RW/1C	1'h0	**Continuous Mode Command Complete for Ns Samples (CONT_CC)** When this register is read, a value of 1 indicates that an ADC command complete for Ns samples in Continuous Mode was completed since the bit was last reset. When this register is written, a value of 1 clears the bit.
1	RW/1C	1'h0	**FO - FIFO Overrun (FO)** When read, a 1 indicates a FIFO overrun. When written, a 1 clears the bit.
0	RW/1C	1'h0	**CC - Command Complete except for Continuous Mode (CC)** When read, a 1 indicates that an ADC command except for Continuous Mode was completed since the bit was last reset. When written, a 1 clears the bit.

22.4.1.4 ADC Interrupt Enable (ADC_INTR_ENABLE)

MEM Offset (00000000)	28h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:3	RO	29'h0000_0000	**RSVD (RSVD)** Reserved
2	RW	1'h0	**CONT_CC - Continuous Mode Command Complete (CONT_CC)** 1 enables the interrupt.
1	RW	1'h0	**FO - FIFO Overrun (FO)** 1 enables the interrupt
0	RW	1'h0	**CC - Command Complete (CC)** 1 enables the interrupt.

22.4.1.5 ADC Sample Register (ADC_SAMPLE)

MEM Offset (00000000)	2Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**ADC Sample (Sample)** when read, this bit-field contains the ADC sample. A read unloads one sample from the FIFO. The FIFO is flushed on a write.

22.4.1.6 ADC Calibraton Data Register (ADC_CALIBRATION)

MEM Offset (00000000)	30h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:7	RO	25'h0000000	**RSVD (RSVD)** Reserved
6:0	RO	7'h0	**CDO- Calibration Data Output (CDO)** When read, this bit field contains the ADC digital offset calibration data output from ADC after calibration completion.

22.4.1.7 ADC FIFO Count Register (ADC_FIFO_COUNT)

MEM Offset (00000000)	34h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:6	RO	26'h0000000	**RSVD (RSVD)** Reserved
5:0	RO	6'h0	**ADC FIFO Count (FifoCount)** When read, this bit-field contains the number of samples stored in the ADC FIFO. Writing has no effect.

22.4.1.8 ADC Operating Mode Register (ADC_OP_MODE)

MEM Offset (00000000)	38h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0FA0h

Bits	Access Type	Default	Description
31:28	RO	4'h0	**RSVD (RSVD)** Reserved
27	RW	1'h0	**Interrupt Enable (IE)** Writing a 1 to this bit enables an interrupt at the completion of the operating mode change. Reading this bit returns the current interrupt enable setting.
26:24	RW	3'h0	**Enctrl - Enable Direct Control (Enctrl)** Writing a non-zero value places the ADC into special test mode. Set this field to zero for normal operation.
23:16	RO	8'h00	**RSVD (RSVD)** Reserved
15:3	RW	13'd500	**Delay (Delay)** Writing to this bit field along with operating mode sets the delay between mode transitions in system clock cycles. Reading this bit field returns the current delay value.
2:0	RW	3'h0	**OM - Operating Mode (OM)** Writing to this bit field along with delay initiates a change in the operating mode. Reading this bit field returns the current operating mode. It is encoded thusly: 0 = Deep power down 1 = Power down 2 = Standby 3 = Normal w/calibration 4 = Normal wo/calibration 5 = No change 6 = No change 7 = No change

§

23 Interrupt Routing

The Interrupt Routing consists of several elements:

1. Internal Host Processor Interrupts
2. SoC Interrupts with configurable routing to Processor
3. Capability for SoC Interrupts to trigger a Processor Halt for Debug
4. Capability for SoC Interrupts to trigger a Warm Reset

23.1 Interrupt Routing

23.1.1 Host Processor Interrupts

The Interrupt Vector Assignments for the Host Processor are described in Table 61:

Table 61. Host Processor Interrupt Vector Assignments

Vector No.	Description	Type
0	Divide Error	Exception
1	Debug Exception	Exception/Trap
2	NMI Interrupt	Interrupt
3	Breakpoint	Trap
4	Overflow	Trap
5	BOUND Range Exceeded	Exception
6	Invalid Opcode	Exception
7	Device Not Available	Exception
8	Double Fault	Abort
9	Intel Reserved.	N/A
10	Invalid TSS	Exception
11	Segment Not Present	Exception
12	Stack-Segment Fault	Exception
13	General Protection Fault	Exception
14	Intel Reserved.	N/A
15	Intel Reserved.	N/A
16	Intel Reserved.	N/A
17	Alignment Check	Exception
18-31	Intel Reserved.	N/A

Interrupt Vectors 0 to 31 are due to events internal to the Host Processor, interrupts due to SoC events are routed through the User Defined Interrupts.

The User Defined Interrupts are delivered to the Host Processor via the APIC which maps particular Interrupt Inputs (IRQs) to configured Interrupt Vectors before presenting the interrupt to the Processor. The SoC Interrupt table shows the IRQ number into the APIC rather than the hardwired Interrupt Vector.

23.1.2 SoC Interrupts and Routing

Unused IRQs (63-51) in Table 62 are Reserved. Interrupts have fixed high priority. Higher the IRQ number, higher is its priority.

Table 62. SoC Interrupt List and Routing Capability

Interrupt Source	Interrupt Source	Description	Host Processor IRQ No.	Host Processor Halt for Debug
I2C MST 0	I2C MST 0	I2C Master 0 Single Interrupt	36	Y
SPI MST 0	SPI MST 0	SPI Master 0 Single Interrupt	39	Y
SPI SLV	SPI SLV	SPI Slave Single Interrupt	37	Y
UART 0	UART 0	UART 0 Single Interrupt	40	Y
UART 1	UART 1	UART 1 Single Interrupt	38	Y
GPIO	GPIO	GPIO Single Interrupt	47	Y
Timer	Timer	Timer Single Interrupt	43	Y
RTC	RTC	RTC Single Interrupt	34	Y
Watchdog	Watchdog	Watchdog Interrupt	48	Y
DMA Channel 0	DMA Channel 0	DMA Channel 0 Single Interrupt	45	Y
DMA Channel 1	DMA Channel 1	DMA Channel 1 Single Interrupt	44	Y
Comparators [18:0]	Comparators [18:0]	19 Comparators Routed to Single Interrupt with 19bit Mask per Destination	46	Y
System	System	D2000 Host AHB Bus Error	33	Y
DMA Error[1:0]	DMA Error[1:0]	2 DMA Channel Error Interrupts Routed to Single Interrupt with 8bit Mask per Destination	32	Y

Interrupt Source	Interrupt Source	Description	Host Processor IRQ No.	Host Processor Halt for Debug
Int. SRAM Controller	Int. SRAM Controller	Internal SRAM Memory Protection Error Single Interrupt	49	Y
Int. Flash Controller 0	Int. Flash Controller 0	Internal Flash Controller 0 Memory Protection Error or Program/Erase completion Single Interrupt	50	Y
AON Timer	AON Timer	Always-On Timer Interrupt	35	Y
ADC Power	ADC Power	ADC power sequence done	51	Y
ADC Calibration	ADC Calibration	ADC Conversion/Calibration done	41	Y
LVTTIMER	LVTTIMER	LVT Timer inside D2000 CPU (connected internal to CPU)	42	Y

§

24 System Control Subsystem

The System Control Subsystem (SCSS) contains functional blocks that control power sequencing, clock generation, reset generation, interrupt routing and pin muxing.

24.1 Features

The following is a list of the System Control Subsystem (SCSS) features:

- Clock Generation and Control
- Reset Generation
- Interrupt Routing
- Pin Mux Control
- SoC Configuration Registers
- Always-On Counter
- Always-On Periodic Timer

24.2 Memory Mapped IO Registers

Registers listed are for the SCSS, starting at base address B0800000h.

Table 63. Summary of SCSS Registers—0xB0800000

MEM Address	Default	Instance Name	Name
0xB0800000	0000_0000h	OSC0_CFG0	Hybrid Oscillator Configuration 0
0xB0800004	0000_0000h	OSC0_STAT1	Hybrid Oscillator status 1
0xB0800008	0000_0302h	OSC0_CFG1	Hybrid Oscillator configuration 1
0xB080000C	0000_0000h	OSC1_STAT0	RTC Oscillator status 0
0xB0800010	0000_0000h	OSC1_CFG0	RTC Oscillator Configuration 0
0xB0800018	00CF_7DB5h	CCU_PERIPH_CLK_GATE_CTL	Peripheral Clock Gate Control
0xB080001C	0000_0001h	CCU_PERIPH_CLK_DIV_CTL0	Peripheral Clock Divider Control 0
0xB0800020	0000_0003h	CCU_GPIO_DB_CLK_CTL	Peripheral Clock Divider Control 1
0xB0800024	0000_0007h	CCU_EXT_CLOCK_CTL	External Clock Control Register
0xB080002C	0001_F000h	CCU_LP_CLK_CTL	System Low Power Clock Control
0xB0800030	FFFF_FFFFh	WAKE_MASK	Wake Mask register
0xB0800034	0000_0014h	CCU_MLAYER_AHB_CTL	AHB Control Register
0xB0800038	0000_0087h	CCU_SYS_CLK_CTL	System Clock Control Register
0xB080003C	0000_0000h	OSC_LOCK_0	Clocks Lock Register

MEM Address	Default	Instance Name	Name
0xB0800040	0000_0000h	SOC_CTRL	SoC Control Register
0xB0800044	0000_0000h	SOC_CTRL_LOCK	SoC Control Register Lock
0xB0800100	0000_0000h	GPS0	General Purpose Sticky Register 0
0xB0800104	0000_0000h	GPS1	General Purpose Sticky Register 1
0xB0800108	0000_0000h	GPS2	General Purpose Sticky Register 2
0xB080010C	0000_0000h	GPS3	General Purpose Sticky Register 3
0xB0800114	0000_0000h	GP0	General Purpose Scratchpad Register 0
0xB0800118	0000_0000h	GP1	General Purpose Scratchpad Register 1
0xB080011C	0000_0000h	GP2	General Purpose Scratchpad Register 2
0xB0800120	0000_0000h	GP3	General Purpose Scratchpad Register 3
0xB0800130	0000_0000h	WO_SP	Write-Once Scratchpad Register
0xB0800134	0000_0000h	WO_ST	Write Once Sticky Register
0xB0800300	0000_0000h	CMP_EN	Comparator enable
0xB0800304	0000_0000h	CMP_REF_SEL	Comparator reference select
0xB0800308	0000_0000h	CMP_REF_POL	Comparator reference polarity select register
0xB080030C	0000_0000h	CMP_PWR	Comparator power enable register
0xB0800328	0000_0000h	CMP_STAT_CLR	Comparator clear register
0xB0800448	0001_0001h	INT_I2C_MST_0_MASK	Host Processor Interrupt Routing Mask 0
0xB0800454	0001_0001h	INT_SPI_MST_0_MASK	Host Processor Interrupt Routing Mask 2
0xB080045C	0001_0001h	INT_SPI_SLV_MASK	Host Processor Interrupt Routing Mask 4
0xB0800460	0001_0001h	INT_UART_0_MASK	Host Processor Interrupt Routing Mask 5
0xB0800464	0001_0001h	INT_UART_1_MASK	Host Processor Interrupt Routing Mask 6
0xB080046C	0001_0001h	INT_GPIO_MASK	Host Processor Interrupt Routing Mask 8
0xB0800470	0001_0001h	INT_TIMER_MASK	Host Processor Interrupt Routing Mask 9
0xB0800478	0001_0001h	INT_RTC_MASK	Host Processor Interrupt Routing Mask 11
0xB080047C	0001_0001h	INT_WATCHDOG_MASK	Host Processor Interrupt Routing Mask 12
0xB0800480	0001_0001h	INT_DMA_CHANNEL_0_MASK	Host Processor Interrupt Routing Mask 13
0xB0800484	0001_0001h	INT_DMA_CHANNEL_1_MASK	Host Processor Interrupt Routing Mask 14
0xB08004A8	0007_FFFFh	INT_COMPARATORS_HOST_HALT_MASK	Host Processor Interrupt Routing Mask 23
0xB08004B0	0007_FFFFh	INT_COMPARATORS_HOST_MASK	Host Processor Interrupt Routing Mask 25
0xB08004B4	0001_0001h	INT_HOST_BUS_ERR_MASK	Host Processor Interrupt Routing Mask 26
0xB08004B8	0003_0003h	INT_DMA_ERROR_MASK	Host Processor Interrupt Routing Mask 27
0xB08004BC	0001_0001h	INT_SRAM_CONTROLLER_MASK	Host Processor Interrupt Routing Mask 28
0xB08004C0	0001_0001h	INT_FLASH_CONTROLLER_0_MASK	Host Processor Interrupt Routing Mask 29

MEM Address	Default	Instance Name	Name
0xB08004C8	0001_0001h	INT_AON_TIMER_MASK	Host Processor Interrupt Routing Mask 31
0xB08004CC	0001_0001h	INT_ADC_PWR_MASK	Host Processor Interrupt Routing Mask 32
0xB08004D0	0001_0001h	INT_ADC_CALIB_MASK	Host Processor Interrupt Routing Mask 33
0xB08004D8	0000_0000h	LOCK_INT_MASK_REG	Interrupt Mask Lock Register
0xB0800540	0000_0010h	AON_VR	AON Voltage Regulator
0xB0800558	0000_000Fh	PM_WAIT	Power Management Wait
0xB0800560	0000_0000h	P_STS	Processor Status
0xB0800570	0000_0000h	RSTC	Reset Control
0xB0800574	0000_0000h	RSTS	Reset Status
0xB0800594	0000_0000h	PM_LOCK	Power Management Lock
0xB0800700	0000_0000h	AONC_CNT	Always on counter register
0xB0800704	0000_0001h	AONC_CFG	Always on counter enable
0xB0800708	0000_0000h	AONPT_CNT	Always on periodic timer
0xB080070C	0000_0001h	AONPT_STAT	Always on periodic timer status register
0xB0800710	0000_0000h	AONPT_CTRL	Always on periodic timer control
0xB0800714	0000_0000h	AONPT_CFG	Always on periodic timer configuration register
0xB0800804	0000_0000h	PERIPH_CFG0	Peripheral Configuration
0xB0800810	0000_0000h	CFG_LOCK	Configuration Lock
0xB0800900	00D0_0000h	PMUX_PULLUP	Pin Mux Pullup
0xB0800910	0000_0000h	PMUX_SLEW	Pin Mux Slew Rate
0xB0800920	03FF_FFFFh	PMUX_IN_EN	Pin Mux Input Enable
0x930	0000_0000h	PMUX_SEL[0]	Pin Mux Select
0x934	0000_0000h	PMUX_SEL[1]	Pin Mux Select
0xB080094C	0000_0000h	PMUX_PULLUP_LOCK	Pin Mux Pullup Lock
0xB0800950	0000_0000h	PMUX_SLEW_LOCK	Pin Mux Slew Rate Lock
0xB0800954	0000_0000h	PMUX_SEL_0_LOCK	Pin Mux Select Lock 0
0xB0800960	0000_0000h	PMUX_IN_EN_LOCK	Pin Mux Slew Rate Lock
0xB0801000	0000_0010h	ID	Identification Register
0xB0801004	0000_0000h	REV	Revision Register
0xB0801008	0000_0024h	FS	Flash Size Register
0xB080100C	0000_0008h	RS	RAM Size Register
0xB0801010	0000_0008h	COTPS	Code OTP Size Register
0xB0801014	0000_0004h	DOTPS	Data OTP Size Register

24.3 Register Detailed Description

24.3.1.1 Hybrid Oscillator Configuration 0 (OSC0_CFG0)

MEM Offset (00000000)	0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:24	RW/P/L	8'h0	**Test Mode Inputs (OSC0_HYB_SET_REG4)** OSC0_CFG0[31:28]: Unused OSC0_CFG0[27]: 0b: Gate output of oscillator with LOCK signal 1b: Give out oscillator output directly OSC0_CFG0[26:25]: 00b: Silicon oscillator counter test mode bits 32M/16M/8M count value = 44/22/6(default) 01b: Count value=68/34/17 10b: Count value=24/12/3 11b: No output
23:16	RW/P/L	8'h0	**Test Mode Inputs (OSC0_HYB_SET_REG3)** OSC0_CFG0[23]: 0b: Default decrease RDAC code for retention mode 1b: Disable this option and feed the same code OSC0_CFG0[22:21]: 00b: crystal oscillator counter test bits; Count value: 3327 (default) 01b: Count value: 1279 10b: Count value: 7423 11b: Count value: 5375 OSC0_CFG0[20:19]: 00b: Default bias current to Gm mos in crystal oscillator 01b: Increase bias current by 23% 10b: Decrease bias current by 23% 11b: Increase bias current by 46% OSC0_CFG0[18:17]: 00b: Default fixed cap in cap trim array crystal oscillator

Bits	Access Type	Default	Description
			01b: Increase fixed cap by 19% 10b: Decrease fixed cap by 38% 11b: Increase fixed cap by 38% OSC0_CFG0[16]: Unused
15:8	RW/P/L	8'h0	**Test Mode Inputs (OSC0_HYB_SET_REG2)** OSC0_CFG0[15:14]: 00b: Default POR generation in silicon oscillator 01b: Delay POR generation in silicon oscillator 10b: Generate POR signal earlier than default 11b: Default start up of PTAT current reference OSC0_CFG0[13:12]: 00b: Default bias current to biasgen block for VREF generation 01b: Zero bias current 10b: Make the bias current 1.5 times its default value 11b: Make the bias current 0.5 times its default value OSC0_CFG0[11:10]: 00b: Default bias current to biasgen block amp 01b: Zero bias current 10b: Make the bias current 1.5 times its default value 11b: Make the bias current 0.5 times its default value OSC0_CFG0[9:8]: 00b: Default bias current to comparator 01b: Zero bias current 10b: Make the bias current 1.5 times its default value 11b: Make the bias current 0.5 times its default value
7:0	RW/P/L	8'h0	**Test Mode Inputs (OSC0_HYB_SET_REG1)** OSC0_CFG0[7]: 0b: Default Gate output of crystal oscillator with LOCK 1b: Bypass LOCK and give out clock output directly OSC0_CFG0[6]: 0b: Default start up of PTAT current reference 1b: External start up for PTAT current reference OSC0_CFG0[5]: 0b: Default start up of compensated current reference 1b: External start up for compensated current reference

Bits	Access Type	Default	Description
			OSC0_CFG0[4]: 0b: Default start up of PTAT current reference 1b: Select external start up for PTAT current reference OSC0_CFG0[3]: 0b: Default start up of compensated current reference 1b: Select external start up for compensated current reference OSC0_CFG0[2]: 0b: Default cut start up current of both current references 1b: Disable cutting the start up branch current OSC0_CFG0[1]: 0b: Normal operation of silicon oscillator 1b: Trimming operation of silicon oscillator OSC0_CFG0[0]: 0b: silicon oscillator works in normal mode (supply = 1.8V) 1b: silicon oscillator works in retention mode (supply = 1.2V)

24.3.1.2 Hybrid Oscillator status 1 (OSC0_STAT1)

MEM Offset (00000000)	4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'h0000_0000	**RSVD (RSVD)** Reserved
1	RO/V	1'h0	**Oscillator 0 Lock (OSC0_LOCK_XTAL)** High output indicates crystal oscillator output is stable.
0	RO/V	1'h0	**Oscillator 1 Lock (OSC0_LOCK_SI)** High output indicates silicon oscillator output is stable.

24.3.1.3 Hybrid Oscillator configuration 1 (OSC0_CFG1)

MEM Offset (00000000)	8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0302h

Bits	Access Type	Default	Description	ResetSignal
31:30	RO	2'h0	**RSVD (RSVD)** Reserved	
29:20	RW/P/L	10'h0	**Trim Code (OSC0_FTRIMOTP)** 10 bit trim code from OTP	
19:16	RW/P/L	4'h0	**Crystal oscillator trim bits (OSC0_FADJ_XTAL)** Trim Code corresponding to load capacitance on board. 10 pF is the default load cap and the corresponding default trim code is 0000. 0111b: 5.55 pF 0110b: 6.18 pF 0101b: 6.82 pF 0100b: 7.45 pF 0011b: 8.08 pF 0010b: 8.71 pF 0001b: 9.34 pF 0000b: 10 pF 1111b: 10.61 pF 1110b: 11.24 pF 1101b: 11.88 pF 1100b: 12.51 pF 1011b: 13.14 pF 1010b: 13.77 pF 1001b: 14.4 pF 1000b: 15.03 pF	
15	RO	1'h0	**RSVD (RSVD)** Reserved	

Bits	Access Type	Default	Description	ResetSignal
14:13	RW/P/L	2'h0	**Oscillator 0 Temperature Control (OSC0_TEMPCOMPPRG)** Bits to control the temperature compensation of silicon oscillator. 00b: Default temperature compensation 01b: Default temperature compensation 10b: Reduce PTAT current in temperature compensation 11b: Increase PTAT current in temperature compensation	
12:10	RW/P/L	3'h0	**Oscillator 0 bias current Control (OSC0_IBIASPRG)** Bits to control the bias current of the silicon oscillator.	
9:8	RW/P	2'h3	**Oscillator frequency selection bits (OSC0_SI_FREQ_SEL)** 00b: 32mHz 01b: 16MHz 10b: 8MHz 11b: 4MHz	
7:5	RW/P/L	3'h0	**Oscillator 0 start-up (OSC0_START_UP)** Bits to control the start-up of silicon oscillator core.	
4	RW/P/L	1'h0	**Oscillator 0 Bypass Mode Enable (OSC0_BYP_XTAL)** Port to enable bypass mode for crystal oscillator. 1b: enable 0b: disable	
3	RW/P	1'h0	**Oscillator 0 Mode Select (OSC0_MODE_SEL)** Selects between crystal and silicon oscillator output: 0b: Silicon oscillator output 1b: Crystal oscillator output	
2	RW/P/L	1'h0	**Oscillator 0 Power-down control (OSC0_PD)** 0b: Oscillator in active mode 1b: Oscillator in power down mode	pwr_rst_n
1	RW/P	1'h1	**Silicon Oscillator Enable (OSC0_EN_SI_OSC)** When high, this enables the silicon oscillator.	
0	RW/P	1'h0	**Crystal Oscillator Enable (OSC0_EN_CRYSTAL)** When high, this enables the crystal oscillator.	

24.3.1.4 RTC Oscillator status 0 (OSC1_STAT0)

MEM Offset (00000000)	0Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved
0	RO/V	1'h0	**Oscillator 1 Lock (OSC1_LOCK_XTAL_OSC)** High output indicates CLKOUT of RTC has stabilized.

24.3.1.5 RTC Oscillator Configuration 0 (OSC1_CFG0)

MEM Offset (00000000)	10h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:18	RO	14'h0000	**RSVD (RSVD)** Reserved	
17:10	RW/P/L	8'h0	**Debug mode test bits (OSC1_XTAL_32K_SET_REG2)** OSC1_CFG0[17:10] Unused	
9:2	RW/P/L	8'h0	**Debug mode test bits (OSC1_XTAL_32K_SET_REG1)** OSC1_CFG0[9:7]: 000b: Default bias current to Gm MOS 010b: Increase bias current by 50% 011b: Increase bias current by 75% 110b: Decrease bias current by 50% 111b: Decrease bias current by 25% OSC1_CFG0[6:4]: 000b: Counter TM bits : Default Count :- 9011 Cycles(275ms) 001b: Count 25395 cycles (775ms) 010b: Count 13107 cycles (400ms) 011b: Count 29491 cycles (900ms) 100b: Count 819 cycles (25ms) 101b: Count 17203 cycles (525ms) 110b: Count 4915 cycles (150ms) 111b: Count 21299 cycles (650ms) OSC1_CFG0[3]: 0b: Gate output of oscillator with LOCK signal 1b: Give out oscillator output directly	

Bits	Access Type	Default	Description	ResetSignal
			OSC0_CFG0[2]: Unused	
1	RW/P/L	1'h0	**Power down signal for crystal oscillator (OSC1_PD)** 1b: disable 0b: enable	pwr_rst_n
0	RW/P/L	1'h0	**RTC Enable Bypass mode (OSC1_BYP_XTAL_UP)** 1b: enable 0b: disable	

24.3.1.6 Peripheral Clock Gate Control (CCU_PERIPH_CLK_GATE_CTL)

MEM Offset (00000000)	18h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	00CF_7DB5h

Bits	Access Type	Default	Description
31:24	RO	8'h00	**RSVD (RSVD)** Reserved
23	RW/P	1'b1	**ADC pclk clock gate enable (CCU_ADC_PCLK_EN_SW)** 1b: enable 0b: disable
22	RW/P	1'b1	**ADC Clock Enable (CCU_ADC_CLK_EN)** 1b: enable 0b: disable
21:20	RO	2'h0	**RSVD (RSVD)** Reserved
19	RW/P	1'b1	**I^{2}C master 0 pclk clock gate enable (CCU_I2C_M0_PCLK_EN_SW)** 1b: enable 0b: disable
18	RW/P	1'b1	**UART B pclk clock gate enable (CCU_UARTB_PCLK_EN_SW)** 1b: enable 0b: disable
17	RW/P	1'b1	**UART A pclk clock gate enable (CCU_UARTA_PCLK_EN_SW)** 1b: enable 0b: disable
16	RW/P	1'b1	**SPI slave pclk clock gate enable (CCU_SPI_PCLK_EN_SW)** 1b: enable 0b: disable

Bits	Access Type	Default	Description
15	RO	1'h0	**RSVD (RSVD)** Reserved
14	RW/P	1'b1	**SPI master 0 pclk clock gate enable (CCU_SPI_M0_PCLK_EN_SW)** 1b: enable 0b: disable
13	RW/P	1'b1	**GPIO pclk clock gate enable (CCU_GPIO_PCLK_EN_SW)** 1b: enable 0b: disable
12	RW/P	1'b1	**Timer pclk clock gate enable (CCU_TIMER_PCLK_EN_SW)** 1b: enable 0b: disable
11	RW/P	1'b1	**RTC pclk clock gate enable (CCU_RTC_PCLK_EN_SW)** 1b: enable 0b: disable
10	RW/P	1'b1	**Watch Dog Timer clock gate enable (CCU_WDT_PCLK_EN_SW)** 1b: enable 0b: disable
9	RO	1'h0	**RSVD (RSVD)** Reserved
8	RW/P	1'b1	**General Purpose IO Debounce Clock Enable (CCU_PERIPH_GPIO_DB_CLK_EN)** 1b: enable 0b: disable
7	RW/P	1'b1	**General Purpose IO interrupt Clock Enable (CCU_GPIO_INTR_CLK_EN)** 1b: enable 0b: disable
6	RO	1'h0	**RSVD (RSVD)** Reserved
5	RW/P	1'b1	**SPI Master 0 clock enable (CCU_SPI_M0_CLK_EN)** 1b: enable 0b: disable
4	RW/P	1'b1	**SPI Slave Clock Enable (CCU_SPI_S_CLK_EN)** 1b: enable 0b: disable
3	RO	1'h0	**RSVD (RSVD)** Reserved
2	RW/P	1'b1	**I^{2}C Master 0 Clock enable (CCU_I2C_M0_CLK_EN)** 1b: enable 0b: disable

Bits	Access Type	Default	Description
1	RW/P	1'b0	**Peripheral clock enable (CCU_PERIPH_CLK_EN)** This controls the peripheral clock trunk. By default, clock to peripheral Subsytem is disabled. 1b: enable 0b: disable
0	RW/P	1'b1	**PERIPH_HCLK_EN (CCU_PERIPH_HCLK_EN)** This controls the peripheral hclk clock which is given to periph icm & AHB2APB Bridge. 1b: enable 0b: disable

24.3.1.7 Peripheral Clock Divider Control 0 (CCU_PERIPH_CLK_DIV_CTL0)

MEM Offset (00000000)	1Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:26	RO	6'h00	**RSVD (RSVD)** Reserved
25:16	RW/P	10'h0	**ADC Clock divider (divisor is N+1) (CCU_ADC_CLK_DIV)** 0d: divide by 1 1d: divide by 2 2d: divide by 3 ..... 1023d: divide by 1024
15:3	RO	13'h0000	**RSVD (RSVD)** Reserved
2:1	RW/P	2'h0	**Peripheral Clock divider (CCU_PERIPH_PCLK_DIV)** 00b: divide by 1 01b: divide by 2 10b: divide by 4 11b: divide by 8
0	RW/P	1'b1	**Peripheral Clock divider enable (CCU_PERIPH_PCLK_DIV_EN)** This bit must be written from 0 -> 1 to apply the value.

24.3.1.8 Peripheral Clock Divider Control 1 (CCU_GPIO_DB_CLK_CTL)

MEM Offset (00000000)	20h
Security_PolicyGroup	

IntelRsvd	False
Size	32 bits
Default	0000_0003h

Bits	Access Type	Default	Description
31:5	RO	27'h0000000	**RSVD (RSVD)** Reserved
4:2	RW/P	3'h0	**General Purpose IO debounce clock divider (CCU_GPIO_DB_CLK_DIV)** 000b: divide by 1 001b: divide by 2 010b: divide by 4 011b: divide by 8 100b: divide by 16 101b: divide by 32 110b: divide by 64 111b: divide by 128
1	RW/P	1'b1	**General Purpose IO debounce clock divider enable (CCU_GPIO_DB_CLK_DIV_EN)** This bit must be written from 0 -> 1 to apply the value.
0	RW/P	1'b1	**General Purpose IO Debounce Clock Enable (CCU_GPIO_DB_CLK_EN)** 1b: enable 0b: disable

24.3.1.9 External Clock Control Register (CCU_EXT_CLOCK_CTL)

MEM Offset (00000000)	24h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0007h

Bits	Access Type	Default	Description
31:5	RO	27'h0000000	**RSVD (RSVD)** Reserved
4:3	RW/P	2'h0	**External clock divider (CCU_EXT_CLK_DIV)** 00b: divide by 1 01b: divide by 2 10b: divide by 4 11b: divide by 8
2	RW/P	1'b1	**External clock divider enable (CCU_EXT_CLK_DIV_EN)** This bit must be written from 0 -> 1 to apply the value.

Bits	Access Type	Default	Description
1	RW/P	1'b1	**External Clock Enable (CCU_EXT_CLK_EN)** 1b: enable 0b: disable
0	RW/P	1'b1	**External RTC enable (CCU_EXT_RTC_EN)** 1b: enable 0b: disable

24.3.1.10 System Low Power Clock Control (CCU_LP_CLK_CTL)

MEM Offset (00000000)	2Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_F000h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P	1'b1	**Clock gate Enable for AON_CNT (CCU_AON_TMR_CNT_CLK_EN_SW)** Clock gate Enable for AON_CNT. 1b: Clock is Enabled 0b: Clock is disabled
15	RW/P	1'b1	**Latch Enable for the RTC Osc PD Latch (RTC_OSC_PD_LATCH_EN)** Latch Enable for the RTC Osc PD Latch. 1b: OSC1_CFG0.OSC1_PD register value is passed to RTC Osc immediaetly 0b: OSC1_CFG0.OSC1_PD register value is passed to RTC Osc only when CPU is halted
14	RW/P	1'b1	**Latch Enable for the Hybrid Osc PD Latch (HYB_OSC_PD_LATCH_EN)** Latch Enable for the Hybrid Osc PD Latch. 1b: OSC0_CFG1.OSC0_PD register value is passed to Hybrid Osc immediaetly 0b: OSC0_CFG1.OSC0_PD register value is passed to Hybrid Osc only when CPU is halted
13	RW/P	1'b1	**Wake Mask for Probe Mode (WAKE_PROBE_MODE_MASK)** Wake Mask Enable for Probe Mode irq source. If enabled, probe mode irq source will act as the wake source to exit from low power state. 1b: Wake source is Masked 0b: Wake source is Enabled
12:8	RW/P	5'b10000	**CCU CPU Halt Clock Count (CCU_CPU_HALT_CLK_CNT)** This defines the number of clock cycles clock can be gated whenever cpu executes halt instruction.

Bits	Access Type	Default	Description
7:5	RO	3'h0	**RSVD (RSVD)** Reserved
4	RW/P	1'b0	**CCU LP Exit to Hybrid Oscillator (CCU_EXIT_TO_HYBOSC)** This is provided to control the mux between hybrid oscillator and RTC oscillator for sys_clk at the time of exiting low power state. 0b : No operation. sys_clk mux is controlled by CCU_SYS_CLK_SEL. 1b : CCU_SYS_CLK_SEL is overridden to select hybrid oscillator at the time of low power state exit when wake event occurs.
3	RW/P	1'b0	**CCU Memory Subsystem Halt Enable (CCU_MEM_HALT_EN)** This defines how the clock to the Memory Subsystem (SRAM, Flash) is controlled in SoC active state. 0b : Clock to SRAM & Flash are controlled by respective SW clock enables. 1b : Clock to SRAM & Flash runs whenever the clock to host processor is running. In this mode, SW clock enables are ignored.
2	RW/P	1'b0	**CCU CPU Halt Enable (CCU_CPU_HALT_EN)** This defines how the clock to the Host Processor Subsystem (CPU core, Local APIC and IOAPIC) is controlled in SoC active state. 0b : Host Processor Subsytem clock is kept enabled irrespective of halt execution. 1b : Host Processor Subsytem clock is disabled when cpu executes halt instuction. And is reenabled when interrupt event occurs.
1:0	RO	2'h0	**RSVD (RSVD)** Reserved

24.3.1.11 Wake Mask register (WAKE_MASK)

MEM Offset (00000000)	30h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	FFFF_FFFFh

Bits	Access Type	Default	Description
31:0	RW/P/L	32'hFFFFFFFF	**Wake Mask register (WAKE_MASK)** Wake Mask Enable for corresponding 32 irq sources. [31:19]: RESERVED [18]: FLASH_CONTROLLER_0 [17]: SRAM_CONTROlIER [16]: WATCHDOG [15]: GPIO [14]: COMPARATORS [13]: DMA_CHANNEL_0 [12]: DMA_CHANNEL_1

Bits	Access Type	Default	Description
			[11]: TIMER [10]: ADC_POWER [9]: ADC_CALIBRATION [8]: UART_0 [7]: SPI_MST_0 [6]: UART_1 [5]: SPI_SLV [4]: I2C_MST_0 [3]: AON_TIMER [2]: RTC [1]: HOST_BUS_ERR [0]: DMA_ERROR If enabled, respective irq source will act as the wake source to exit from low power state. 1b: Wake source is Masked 0b: Wake source is Enabled

24.3.1.12 AHB Control Register (CCU_MLAYER_AHB_CTL)

MEM Offset (00000000)	34h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0014h

Bits	Access Type	Default	Description
31:7	RO	25'h0000000	**RSVD (RSVD)** Reserved
6	RW/P	1'b0	**DMA Clock enable (CCU_DMA_CLK_EN)** This controls clock to DMA Controller. By default clock to DMA controller is disabled. 1b: enable 0b: disable
5	RO	1'h0	**RSVD (RSVD)** Reserved
4	RW/P	1'b1	**SRAM Clock Enable (CCU_SRAM_CLK_EN)** 1b: enable 0b: disable
3	RO	1'h0	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
2	RW/P	1'b1	**FLASH 0 Clock Enable (CCU_FLASH0_CLK_EN)** 1b: enable 0b: disable
1:0	RO	2'h0	**RSVD (RSVD)** Reserved

24.3.1.13 System Clock Control Register (CCU_SYS_CLK_CTL)

MEM Offset (00000000)	38h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0087h

Bits	Access Type	Default	Description
31:11	RO	21'h000000	**RSVD (RSVD)** Reserved
10:8	RW/P	3'h0	**System Clock Divider (CCU_SYS_CLK_DIV)** 000b: divide by 1 001b: divide by 2 010b: divide by 4 011b: divide by 8 100b: divide by 16 101b: divide by 32 110b: divide by 64 111b: divide by 128
7	RW/P	1'b1	**System Clock Divider Enable (CCU_SYS_CLK_DIV_EN)** This bit must be written from 0 -> 1 to apply the value.

Bits	Access Type	Default	Description
6:3	RW/P/L	4'h0	**RTC Clock Divider (CCU_RTC_CLK_DIV)** 0000b: divide by 1 0001b: divide by 2 0010b: divide by 4 0011b: divide by 8 0100b: divide by 16 0101b: divide by 32 0110b: divide by 64 0111b: divide by 128 1000b: divide by 256 1001b: divide by 512 1010b: divide by 1024 1011b: divide by 2048 1100b: divide by 4096 1101b: divide by 8192 1110b: divide by 16384 1111b: divide by 32768
2	RW/P/L	1'b1	**RTC Clock Divider Enable (CCU_RTC_CLK_DIV_EN)** This bit must be written from 0 -> 1 to apply the value.
1	RW/P/L	1'b1	**RTC Clock Enable (CCU_RTC_CLK_EN)** 1b: enable 0b: disable
0	RW/P	1'b1	**Select Clock (CCU_SYS_CLK_SEL)** 0b: 32 kHz RTC Crystal Oscillator 1b: 32 MHz Hybrid Oscillator

24.3.1.14 Clocks Lock Register (OSC_LOCK_0)

MEM Offset (00000000)	3Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31	RW/1S	1'h0	**Test Mode Input 4 Lock (OSC0_HYB_SET_REG4_LOCK)** 1b: Lock	pwr_rst_n
30	RW/1S	1'h0	**Test Mode Input 3 Lock (OSC0_HYB_SET_REG3_LOCK)** 1b: Lock	pwr_rst_n
29	RW/1S	1'h0	**Test Mode Input 2 Lock (OSC0_HYB_SET_REG2_LOCK)** 1b: Lock	pwr_rst_n

Bits	Access Type	Default	Description	ResetSignal
28	RW/1S	1'h0	**Test Mode Input 1 Lock (OSC0_HYB_SET_REG1_LOCK)** 1b: Lock	pwr_rst_n
27	RW/1S	1'h0	**10 bit trim code from OTP Lock (OSC0_FTRIMOTP_LOCK)** 1b: Lock	pwr_rst_n
26	RW/1S	1'h0	**Crystal oscillator trim bits Lock (OSC0_FADJ_XTAL_LOCK)** 1b: Lock	pwr_rst_n
25	RW/1S	1'h0	**Oscillator 0 Temperature Control Lock (OSC0_TEMPCOMPPRG_LOCK)** 1b: Lock	pwr_rst_n
24	RW/1S	1'h0	**Oscillator 0 bias current Control Lock (OSC0_IBIASPRG_LOCK)** 1b: Lock	pwr_rst_n
23	RW/1S	1'h0	**Oscillator 0 start-up Lock (OSC0_START_UP_LOCK)** 1b: Lock	pwr_rst_n
22	RW/1S	1'h0	**Oscillator 0 Bypass Mode Enable Lock (OSC0_BYP_XTAL_LOCK)** 1b: Lock	pwr_rst_n
21	RO	1'h0	**RSVD (RSVD)** Reserved	
20	RW/1S	1'h0	**Debug mode test bits Reg 2 Lock (OSC1_XTAL_32K_SET_REG2_LOCK)** 1b: Lock	pwr_rst_n
19	RW/1S	1'h0	**Debug mode test bits Reg 1 Lock (OSC1_XTAL_32K_SET_REG1_LOCK)** 1b: Lock	pwr_rst_n
18	RW/1S	1'h0	**RTC Enable Bypass mode Lock (OSC1_BYP_XTAL_UP_LOCK)** 1b: Lock	pwr_rst_n
17:14	RO	4'h0	**RSVD (RSVD)** Reserved	
13	RW/1S	1'h0	**RTC Divider Lock (RTC_DIV_LOCK)** 1b: Lock	pwr_rst_n
12	RW/1S	1'h0	**OSC1 Powerdown Lock (OSC1_PD_LOCK)** 1b: Lock	pwr_rst_n
11	RW/1S	1'h0	**OSC0 Powerdown Lock (OSC0_PD_LOCK)** 1b: Lock	pwr_rst_n
10:0	RO	11'h000	**RSVD (RSVD)** Reserved	

24.3.1.15 SoC Control Register (SOC_CTRL)

MEM Offset (00000000)	40h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b0	**Override OTP Security (OVR_OTP_SEC)** If 8KB OTP area is OTPed, then JTAG access to HVM tap, D2000 tap is not allowed (Intel Orange access). This is a lockable register bit after cold reset, which is written by boot FW to override OTP bit (On-time-programmed bit in Flash) based security restrictions (example: Disabling JTAG access to Flash). 0b: Security restrictions based on OTP bit is enforced. 1b: Override OTP bit based security restrictions

24.3.1.16 SoC Control Register Lock (SOC_CTRL_LOCK)

MEM Offset (00000000)	44h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved	
0	RW/1S	1'b0	**Override OTP Security Lock (OVR_OTP_SEC_LOCK)** Lock OVR_OTP_SEC Field in OVR_OTP_SEC register.	pwr_rst_n

24.3.1.17 General Purpose Sticky Register 0 (GPS0)

MEM Offset (00000000)	100h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/P	32'h0	**Sticky Scratchpad 0 (GPS0)** Cleared on POR or COLD reset.

24.3.1.18 General Purpose Sticky Register 1 (GPS1)

MEM Offset (00000000)	104h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/P	32'h0	**Sticky scratchpad 1 (GPS1)** Cleared on POR or COLD reset.

24.3.1.19 General Purpose Sticky Register 2 (GPS2)

MEM Offset (00000000)	108h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/P	32'h0	**Sticky scratchpad 2 (GPS2)** Cleared on POR or COLD reset.

24.3.1.20 General Purpose Sticky Register 3 (GPS3)

MEM Offset (00000000)	10Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/P	32'h0	**Sticky scratchpad 3 (GPS3)** Cleared on POR or COLD reset.

24.3.1.21 General Purpose Scratchpad Register 0 (GP0)

MEM Offset (00000000)	114h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**General Purpose Scratchpad Register 0 (GP0)** Cleared on POR or COLD or WARM reset.

24.3.1.22 General Purpose Scratchpad Register 1 (GP1)

MEM Offset (00000000)	118h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**General Purpose Scratchpad Register 1 (GP1)** Cleared on POR or COLD or WARM reset.

24.3.1.23 General Purpose Scratchpad Register 2 (GP2)

MEM Offset (00000000)	11Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**General Purpose Scratchpad Register 2 (GP2)** Cleared on POR or COLD or WARM reset.

24.3.1.24 General Purpose Scratchpad Register 3 (GP3)

MEM Offset (00000000)	120h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW	32'h0	**General Purpose Scratchpad Register 3 (GP3)** Cleared on POR or COLD or WARM reset.

24.3.1.25 Write-Once Scratchpad Register (WO_SP)

MEM Offset (00000000)	130h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/1S	32'h0	**Write-Once Scratchpad Register (WO_SP)** This field provides Read/Write-One-to-Set behavior on a per bit basis.

24.3.1.26 Write Once Sticky Register (WO_ST)

MEM Offset (00000000)	134h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:0	RW/1S	32'h0	**Write Once Sticky Register (WO_ST)** This field provides sticky Read/Write-One-to-Set behavior on a per bit basis.	pwr_rst_n

24.3.1.27 Comparator enable (CMP_EN)

MEM Offset (00000000)	300h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18:0	RW/P	19'h0	**Comparator enable (CMP_EN)** If en: 0b, comparator interrupt will not fire.

24.3.1.28 Comparator reference select (CMP_REF_SEL)

MEM Offset (00000000)	304h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18:0	RW/P	19'h0	**Comparator reference select (CMP_REF_SEL)** 0b: Selects Reference from external voltage reference (AR pin). 1b: Selects Reference from internal voltage reference output of voltage regulator (0.95V typ).

24.3.1.29 Comparator reference polarity select register (CMP_REF_POL)

MEM Offset (00000000)	308h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18:0	RW/P	19'h0	**Comparator reference polarity select register (CMP_REF_POL)** 0b: Comparator monitors if Analog input voltage is greater than voltage reference. 1b: Comparator monitors if Analog input voltage is lesser than voltage reference.

24.3.1.30 Comparator power enable register (CMP_PWR)

MEM Offset (00000000)	30Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
18:0	RW/P	19'h0	**Comparator power enable register (CMP_PWR)** Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: Normal Operation mode. 0b: Power-Down/Shutdown mode.

24.3.1.31 Comparator clear register (CMP_STAT_CLR)

MEM Offset (00000000)	328h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_18)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
17	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_17)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
16	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_16)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
15	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_15)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear

Bits	Access Type	Default	Description
14	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_14)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
13	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_13)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
12	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_12)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
11	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_11)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
10	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_10)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
9	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_9)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
8	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_8)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear

Bits	Access Type	Default	Description
7	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_7)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
6	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_6)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
5	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_5)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
4	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_4)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
3	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_3)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
2	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_2)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear
1	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_1)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear

Bits	Access Type	Default	Description
0	RW/1C/V/P	1'h0	**Comparator status clear register (CMP_STAT_CLR_0)** The current status of the latched value of the comparator. Software must clear the latch before another interrupt. Each Bit of This Register Manages Over One Out of Nineteen Comparators. For Each Comparator: 1b: clear

24.3.1.32 Host Processor Interrupt Routing Mask 0 (INT_I2C_MST_0_MASK)

MEM Offset (00000000)	448h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**I^{2}C Master 0 Host Halt interrupt mask (INT_I2C_MST_0_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**I^{2}C Master 0 Host interrupt mask (INT_I2C_MST_0_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.33 Host Processor Interrupt Routing Mask 2 (INT_SPI_MST_0_MASK)

MEM Offset (00000000)	454h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved

Bits	Access Type	Default	Description
16	RW/P/L	1'b1	**SPI Master 0 interrupt mask (INT_SPI_MST_0_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**SPI Master 0 Halt interrupt mask (INT_SPI_MST_0_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.34 Host Processor Interrupt Routing Mask 4 (INT_SPI_SLV_MASK)

MEM Offset (00000000)	45Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**SPI Slave Host Halt interrupt mask (INT_SPI_SLV_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**SPI Slave Host interrupt mask (INT_SPI_SLV_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.35 Host Processor Interrupt Routing Mask 5 (INT_UART_0_MASK)

MEM Offset (00000000)	460h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**UART0 Master Halt interrupt mask (INT_UART_0_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**UART0 Master interrupt mask (INT_UART_0_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.36 Host Processor Interrupt Routing Mask 6 (INT_UART_1_MASK)

MEM Offset (00000000)	464h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**UART1 Master Halt interrupt mask (INT_UART_1_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**UART1 Master interrupt mask (INT_UART_1_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.37 Host Processor Interrupt Routing Mask 8 (INT_GPIO_MASK)

MEM Offset (00000000)	46Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**General Purpose I/O Host Halt interrupt Mask (INT_GPIO_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**General Purpose I/O host interrupt Mask (INT_GPIO_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.38 Host Processor Interrupt Routing Mask 9 (INT_TIMER_MASK)

MEM Offset (00000000)	470h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**Timer Host Halt interrupt mask (INT_TIMER_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**Timer Host interrupt mask (INT_TIMER_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.39 Host Processor Interrupt Routing Mask 11 (INT_RTC_MASK)

MEM Offset (00000000)	478h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**RTC Host Halt interrupt mask (INT_RTC_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**RTC Host interrupt mask (INT_RTC_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.40 Host Processor Interrupt Routing Mask 12 (INT_WATCHDOG_MASK)

MEM Offset (00000000)	47Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**Watchog Host Halt interrupt mask (INT_WATCHDOG_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**Watchog Host interrupt mask (INT_WATCHDOG_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.41 Host Processor Interrupt Routing Mask 13 (INT_DMA_CHANNEL_0_MASK)

MEM Offset (00000000)	480h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**DMA channel 0 Host Halt interrupt mask (INT_DMA_CHANNEL_0_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**DMA channel 0 Host interrupt mask (INT_DMA_CHANNEL_0_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.42 Host Processor Interrupt Routing Mask 14 (INT_DMA_CHANNEL_1_MASK)

MEM Offset (00000000)	484h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**DMA channel 1 Host Halt interrupt mask (INT_DMA_CHANNEL_1_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**DMA channel 1 Host interrupt mask (INT_DMA_CHANNEL_1_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.43 Host Processor Interrupt Routing Mask 23 (INT_COMPARATORS_HOST_HALT_MASK)

MEM Offset (00000000)	4A8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0007_FFFFh

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18:0	RW/P/L	19'h7ffff	**Comparators Host Halt interrupt mask (INT_COMPARATORS_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked

24.3.1.44 Host Processor Interrupt Routing Mask 25 (INT_COMPARATORS_HOST_MASK)

MEM Offset (00000000)	4B0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0007_FFFFh

Bits	Access Type	Default	Description
31:19	RO	13'h0000	**RSVD (RSVD)** Reserved
18:0	RW/P/L	19'h7ffff	**Comparators Host interrupt mask (INT_COMPARATORS_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.45 Host Processor Interrupt Routing Mask 26 (INT_HOST_BUS_ERR_MASK)

MEM Offset (00000000)	4B4h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**Host Processor Bus Error Host Halt mask (INT_HOST_BUS_ERR_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**Host Processor Bus Error Host mask (INT_HOST_BUS_ERR_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.46 Host Processor Interrupt Routing Mask 27 (INT_DMA_ERROR_MASK)

MEM Offset (00000000)	4B8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0003_0003h

Bits	Access Type	Default	Description
31:18	RO	14'h0000	**RSVD (RSVD)** Reserved
17:16	RW/P/L	2'b11	**DMA Error Host Halt Mask - 2 bits (INT_DMA_ERROR_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:2	RO	14'h0000	**RSVD (RSVD)** Reserved
1:0	RW/P/L	2'b11	**DMA Error Host Mask - 2 bits (INT_DMA_ERROR_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.47 Host Processor Interrupt Routing Mask 28 (INT_SRAM_CONTROLLER_MASK)

MEM Offset (00000000)	4BCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**SRAM Controller Host Halt mask (INT_SRAM_CONTROLLER_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**SRAM Controller Host mask (INT_SRAM_CONTROLLER_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.48 Host Processor Interrupt Routing Mask 29 (INT_FLASH_CONTROLLER_0_MASK)

MEM Offset (00000000)	4C0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**Flash Controller 0 Host Halt interrupt mask (INT_FLASH_CONTROLLER_0_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**Flash Controller 0 Host interrupt mask (INT_FLASH_CONTROLLER_0_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.49 Host Processor Interrupt Routing Mask 31 (INT_AON_TIMER_MASK)

MEM Offset (00000000)	4C8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**Always-On Timer Host Halt Interrupt Mask (INT_AON_TIMER_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**Always-On Timer Host Interrupt Mask (INT_AON_TIMER_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.50 Host Processor Interrupt Routing Mask 32 (INT_ADC_PWR_MASK)

MEM Offset (00000000)	4CCh
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**ADC Power int Host Halt Interrupt Mask (INT_ADC_PWR_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**ADC Power int Host Interrupt Mask (INT_ADC_PWR_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.51 Host Processor Interrupt Routing Mask 33 (INT_ADC_CALIB_MASK)

MEM Offset (00000000)	4D0h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0001_0001h

Bits	Access Type	Default	Description
31:17	RO	15'h0000	**RSVD (RSVD)** Reserved
16	RW/P/L	1'b1	**ADC Conversion/Calibration int Host Halt Interrupt Mask (INT_ADC_CALIB_HOST_HALT_MASK)** 0: Interrupt Event triggers a warm reset or entry into probe mode based on P_STS.HALT_INT_REDIR 1: Masked
15:1	RO	15'h0000	**RSVD (RSVD)** Reserved
0	RW/P/L	1'b1	**ADC Conversion/Calibration int Host Interrupt Mask (INT_ADC_CALIB_HOST_MASK)** 0: Interrupt Event triggers interrupt to host processor 1: Masked

24.3.1.52 Interrupt Mask Lock Register (LOCK_INT_MASK_REG)

MEM Offset (00000000)	4D8h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:3	RO	29'h0000_0000	**RSVD (RSVD)** Reserved	
2	RW/1S	1'b0	**Lock All Host Processor Halt Mask Fields (LOCK_HOST_HALT_MASK)**	pwr_rst_n
1	RO	1'h0	**RSVD (RSVD)** Reserved	
0	RW/1S	1'b0	**Lock All Host Processor Interrupt Mask Fields (LOCK_HOST_MASK)**	pwr_rst_n

24.3.1.53 AON Voltage Regulator (AON_VR)

MEM Offset (00000000)	540h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0010h

Bits	Access Type	Default	Description
31:16	WO	16'h0	**Pass Code for AON_VR (VR_PASS_CODE)** When written with a value of 16'h9DC4, the write to this register is performed, any other value inhibits write operation.
15	RO/V	1'h0	**eSR Volatge Regulator status (ROK_BUF_VREG)** 0: eSR Switching Regulator has not attained 1.8V regulation. 1: eSR Switching Regulator has attained 1.8V regulation.
14:10	RO	5'h00	**RSVD (RSVD)** Reserved
9	RW/P/L	1'h0	**Mask for ROK_BUF_VREG. (ROK_BUF_VREG_MASK)** 0: No Mask. 1: Mask ROK_BUF_VREG.
8	RW/P/L	1'h0	**Volatge Regulator Select during low power state. (VREG_SEL)** 0: eSR Switching Regulator is kept enabled during low power state. 1: eSR Switching Regulator disabled and qLR Linear Regulator enabled (retention mode)
7:6	RO	2'h0	**RSVD (RSVD)** Reserved
5	RW/P/L	1'h0	**Voltage Select Strobe (VSTRB)** Enables output voltage programming using VSEL, minimum high pulse width requirement is 1us. The VSEL value is strobed only on a rising edge of VSTRB
4:0	RW/P/L	5'b10000	**Voltage Select (VSEL)** These control bits are used to program the output voltage in conjunction with VSTRB 01000b : 1.20V 01001b : 1.25V 01010b : 1.30V 01011b : 1.35V 01100b : 1.40V 01101b : 1.50V 01110b : 1.60V 01111b : 1.70V 10000b : 1.80V 10001b : 1.90V 10010b : 2.00V

Bits	Access Type	Default	Description
			10011b : 2.10V 10100b : 2.20V 10101b : 2.30V 10110b : 2.40V 10111b : 2.50V 11000b : 2.60V 11001b : 2.70V 11010b : 2.80V 11011b : 2.90V 11100b : 3.00V 11101b : 3.10V 11110b : 3.20V 11111b : 3.30V Values between 00000b and 00111b are illegal.

24.3.1.54 Power Management Wait (PM_WAIT)

MEM Offset (00000000)	558h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_000Fh

Bits	Access Type	Default	Description	ResetSignal
31:8	RO	24'h000000	**RSVD (RSVD)** Reserved	
7:0	RW/L	8'h0f	**Voltage Strobe Wait Duration (VSTRB_WAIT)** Determines the number of clock cycles the hardware state machine waits after using voltage strobe to program a new output voltage for the AON regulator. When the voltage value is changed, this field allows time for the new voltage level to be reached before the AON domain is released from reset. The delay is added by loading a counter that runs off a sys_clk and then waiting for the counter to expire. 00h : Wait 0 Clock Cycles 01h : Wait 1 Clock Cycle 02h : Wait 2 Clock Cycles . . FEh : Wait 65,534 Clock Cycles FFh : Wait 65,535 Clock Cycles	por_rst_n

24.3.1.55 Processor Status (P_STS)

MEM Offset (00000000)	560h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:27	RO	5'h00	**RSVD (RSVD)** Reserved
26	RW/P/L	1'h0	**Halt Interrupt Redirection (HALT_INT_REDIR)** When an enabled host halt interrupt occurs, this bit determines if the interrupt event triggers a warm reset or an entry into Probe Mode. 0b : Warm Reset 1b : Probe Mode Entry
25:4	RO	22'h000000	**RSVD (RSVD)** Reserved
3	RW/1C/V	1'h0	**Processor Bus Error Interrupt Status (BUS_ERR)** Interrupt Status bit indicating that an error response has been observed on the processor bus. 0b : No Bus Error Response Observed 1b : Bus Error Response Observed, if bus error is programmed to cause warm reset, this bit gets cleared at warm reset otherwise software has to clear this bit for this interrupt clearing.
2	RW/1C/V/P	1'h0	**Processor Shutdown (SHDWN)** Status bit indicating the processor has issued a Shutdown special cycle. A Shutdown special cycle is generated when the processor incurs a double fault. The processor remains in Shutdown mode until it is reset. This bit is set when the Shutdown special cycle is issued and can only be cleared by software. 0b : No Shutdown Special Cycle Issued 1b : Shutdown Special Cycle Issued
1	RO	1'h0	**RSVD (RSVD)** Reserved
0	RW/1C/V/P	1'h0	**Processor Halt (HLT)** Status bit indicating the processor has entered the Halt state. When a HLT instruction is executed, the processor transitions to the Halt state. 0b : Processor not in Halt State 1b : Processor was in Halt State

24.3.1.56 Reset Control (RSTC)

A write to this register with RSTC.COLD or RSTC.WARM set initiates a reset. Software must only write to one of these bits at a time, else the behavior is undefined.

These bits automatically clear once the reset occurs, so there is no need for software to clear them.

MEM Offset (00000000)	570h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:4	RO	28'h0000000	**RSVD (RSVD)** Reserved
3	RW/P	1'h0	**Cold Reset (COLD)** When this bit is set, the SoC performs a cold reset.
2	RO	1'h0	**RSVD (RSVD)** Reserved
1	RW	1'h0	**Warm Reset (WARM)** When this bit is set, the SoC performs a warm reset.
0	RO	1'h0	**RSVD (RSVD)** Reserved

24.3.1.57 Reset Status (RSTS)

MEM Offset (00000000)	574h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:6	RO	26'h0000000	**RSVD (RSVD)** Reserved
5	RW/1C/V/P	1'h0	**Processor Bus Error (BUS_ERR)** Status bit indicating that an error response has been observed on the processor bus. 0b : No Bus Error Response Observed 1b : Bus Error Response Observed
4	RO	1'h0	**RSVD (RSVD)** Reserved
3	RW/1C/V/P	1'h0	**Host Processor Halt Interrupt Triggered Warm Reset (HOST_HALT_WRST)** When this bit is set, it indicates that an enabled Host Halt interrupt triggered a warm reset.

Bits	Access Type	Default	Description
2	RO	1'h0	**RSVD (RSVD)** Reserved
1	RW/1C/V/P	1'h0	**Watchdog Timer Triggered Warm Reset (WDG_WRST)** When this bit is set, it indicates that Watchdog Timer in the Peripheral block triggered a warm reset.
0	RW/1C/V/P	1'h0	**Software Initiated Warm Reset (SW_WRST)** When this bit is set, it indicates that warm reset was initiated by software writing to RSTC.WARM.

24.3.1.58 Power Management Lock (PM_LOCK)

MEM Offset (00000000)	594h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:18	RO	14'h0000	**RSVD (RSVD)** Reserved	
17	RW/1S	1'h0	**Wake Mask Wait Lock (WAKE_MASK_LOCK)** Lock bit for WAKE_MASK.WAKE_MASK 0b : Unlocked 1b : Locked	pwr_rst_n
16	RW/1S	1'h0	**Power Management Wait Lock (PM_WAIT_LOCK)** Lock bit for PM_WAIT.VSTRB_WAIT 0b : Unlocked 1b : Locked	pwr_rst_n
15:0	RO	16'h0000	**RSVD (RSVD)** Reserved	

24.3.1.59 Always on counter register (AONC_CNT)

MEM Offset (00000000)	700h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO/V/P	32'h0	**Always on count (AONC_CNT)** 32 Bit Counter value

24.3.1.60 Always on counter enable (AONC_CFG)

MEM Offset (00000000)	704h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSV)**
0	RW/P	1'h1	**Always on count enable (AONC_CNT_EN)** 1b: enable 0b: disable

24.3.1.61 Always on periodic timer (AONPT_CNT)

MEM Offset (00000000)	708h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RO/V/P	32'h0	**Periodic Always On Timer (AONPT_CNT)** 32 Bit Always On Timer Value

24.3.1.62 Always on periodic timer status register (AONPT_STAT)

MEM Offset (00000000)	70Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0001h

Bits	Access Type	Default	Description
31:1	RO	31'h0	**Reserved (RSV)**
0	RO/V/P	1'h1	**Periodic always on timer status (AONPT_STAT)** 1b: always on periodic timer has wrapped and an alarm indication has been generated. This must be cleared before a subsequent alarm indication can be triggered.

24.3.1.63 Always on periodic timer control (AONPT_CTRL)

MEM Offset (00000000)	710h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'h0	**Reserved (RSV)**
1	RW/V/P	1'b0	**Periodic always on timer reset (AONPT_RST)** 1b: always on periodic timer is reset to the value contained in the always on periodic timer configuration register.
0	RW/V/P	1'h0	**Periodic always on timer alarm clear (AONPT_CLR)** 1b is written to this register to clear the periodic always on timer alarm.

24.3.1.64 Always on periodic timer configuration register (AONPT_CFG)

MEM Offset (00000000)	714h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:0	RW/P	32'h0	**Periodic always on time out configuration (AONPT_CFG)** 0000h: Periodic always on counter disabled. When set to a non 0000h value, the periodic always on counter is loaded with the configured time out value and down-counts from the configured value to 0000h. When it reaches 0000h, an interrupt is generated and the counter reloads to the configured time out value.

24.3.1.65 Peripheral Configuration (PERIPH_CFG0)

MEM Offset (00000000)	804h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:2	RO	30'h0000_0000	**RSVD (RSVD)** Reserved
1	RW/P/L	1'h0	**Watchdog Clock Enable (WDT_CLK_EN)** Enables the clock for the peripheral watchdog. 0b : Watchdog Clock Disabled 1b : Watchdog Clock Enabled
0	RW/P/L	1'h0	**Watchdog Speed Up (WDT_SPEED_UP)** Debug mode allowing the watchdog time-out to be accelerated. 0b : Watchdog Speed Up Disabled 1b : Watchdog Speed Up Enabled

24.3.1.66 Configuration Lock (CFG_LOCK)

MEM Offset (00000000)	810h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:2	RO	30'h0000_0000	**RSVD (RSVD)** Reserved	

Bits	Access Type	Default	Description	ResetSignal
1	RW/1S	1'h0	**Watchdog Clock Enable Lock (WDT_CLK_EN_LOCK)** Lock bit for PERIPH_CFG0.WDT_CLK_EN. 0b : Unlocked 1b : Locked	pwr_rst_n
0	RW/1S	1'h0	**Watchdog Speed Up Lock (WDT_SPEED_UP_LOCK)** Lock bit for PERIPH_CFG0.WDT_SPEED_UP. 0b : Unlocked 1b : Locked	pwr_rst_n

24.3.1.67 Pin Mux Pullup (PMUX_PULLUP)

MEM Offset (00000000)	900h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	00D0_0000h

Bits	Access Type	Default	Description
31:25	RO	6'h00	**RSVD (RSVD)** Reserved
24:0	RW/P/L	26'h0D00000	**Pin Mux Pullup Enable (PMUX_PU_EN)** 0b: disable pull-up 1b: enables pull-up(49kOhms typ)

24.3.1.68 Pin Mux Slew Rate (PMUX_SLEW)

MEM Offset (00000000)	910h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:25	RO	6'h00	**RSVD (RSVD)** Reserved
24:0	RW/P/L	26'h0	**Pin mux slew rate (PMUX_SLEW_EN)** 0b: 12 mA driver 1b: 16 mA driver

24.3.1.69 Pin Mux Input Enable (PMUX_IN_EN)

MEM Offset (00000000)	920h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	03FF_FFFFh

Bits	Access Type	Default	Description
31:25	RO	6'h00	**RSVD (RSVD)** Reserved
24:0	RW/P/L	26'h3FFFFFF	**Pin mux input enable (PMUX_IN_EN)** 0b: disable input pad 1b: enables input pad

24.3.1.70 Pin Mux Select (PMUX_SEL [0..1])

MEM Offset (00000000)	[0]:930h [1]:934h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:30	RW/P/L	2'h0	**Pin Mux Select 15 (PMUX_SEL15)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
29:28	RW/P/L	2'h0	**Pin Mux Select 14 (PMUX_SEL14)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
27:26	RW/P/L	2'h0	**Pin Mux Select 13 (PMUX_SEL13)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3

Bits	Access Type	Default	Description
25:24	RW/P/L	2'h0	**Pin Mux Select 12 (PMUX_SEL12)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
23:22	RW/P/L	2'h0	**Pin Mux Select 11 (PMUX_SEL11)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
21:20	RW/P/L	2'h0	**Pin Mux Select 10 (PMUX_SEL10)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
19:18	RW/P/L	2'h0	**Pin Mux Select 9 (PMUX_SEL9)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
17:16	RW/P/L	2'h0	**Pin Mux Select 8 (PMUX_SEL8)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
15:14	RW/P/L	2'h0	**Pin Mux Select 7 (PMUX_SEL7)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
13:12	RW/P/L	2'h0	**Pin Mux Select 6 (PMUX_SEL6)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
11:10	RW/P/L	2'h0	**Pin Mux Select 5 (PMUX_SEL5)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3

Bits	Access Type	Default	Description
9:8	RW/P/L	2'h0	**Pin Mux Select 4 (PMUX_SEL4)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
7:6	RW/P/L	2'h0	**Pin Mux Select 3 (PMUX_SEL3)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
5:4	RW/P/L	2'h0	**Pin Mux Select 2 (PMUX_SEL2)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
3:2	RW/P/L	2'h0	**Pin Mux Select 1 (PMUX_SEL1)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3
1:0	RW/P/L	2'h0	**Pin Mux Select 0 (PMUX_SEL0)** 00b: Select User Mode 0 01b: Select User Mode 1 10b: Select User Mode 2 11b: Select User Mode 3

24.3.1.71 Pin Mux Pullup Lock (PMUX_PULLUP_LOCK)

MEM Offset (00000000)	94Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved	
0	RW/1S	1'h0	**Pin Mux Pullup 0 Enable Lock (PMUX_PU0_EN_LOCK)** 1b: Lock pull-up	pwr_rst_n

24.3.1.72 Pin Mux Slew Rate Lock (PMUX_SLEW_LOCK)

MEM Offset (00000000)	950h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved	
0	RW/1S	1'h0	**Pin Mux Slew Rate 0 Lock (PMUX_SLEW0_EN_LOCK)** 1b: Lock Slew Rate	pwr_rst_n

24.3.1.73 Pin Mux Select Lock 0 (PMUX_SEL_0_LOCK)

MEM Offset (00000000)	954h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31	RW/1S	1'h0	**Pin Mux Select Lock 15 (PMUX_SEL15)** 1b: Lock Pmux Select	pwr_rst_n
30	RW/1S	1'h0	**Pin Mux Select Lock 14 (PMUX_SEL14)** 1b: Lock Pmux Select	pwr_rst_n
29	RW/1S	1'h0	**Pin Mux Select Lock 13 (PMUX_SEL13)** 1b: Lock Pmux Select	pwr_rst_n
28	RW/1S	1'h0	**Pin Mux Select Lock 12 (PMUX_SEL12)** 1b: Lock Pmux Select	pwr_rst_n
27	RW/1S	1'h0	**Pin Mux Select Lock 11 (PMUX_SEL11)** 1b: Lock Pmux Select	pwr_rst_n
26	RW/1S	1'h0	**Pin Mux Select Lock 10 (PMUX_SEL10)** 1b: Lock Pmux Select	pwr_rst_n
25	RW/1S	1'h0	**Pin Mux Select Lock 9 (PMUX_SEL9)** 1b: Lock Pmux Select	pwr_rst_n
24	RW/1S	1'h0	**Pin Mux Select Lock 8 (PMUX_SEL8)** 1b: Lock Pmux Select	pwr_rst_n
23	RW/1S	1'h0	**Pin Mux Select Lock 7 (PMUX_SEL7)** 1b: Lock Pmux Select	pwr_rst_n

Bits	Access Type	Default	Description	ResetSignal
22	RW/1S	1'h0	**Pin Mux Select Lock 6 (PMUX_SEL6)** 1b: Lock Pmux Select	pwr_rst_n
21	RW/1S	1'h0	**Pin Mux Select Lock 5 (PMUX_SEL5)** 1b: Lock Pmux Select	pwr_rst_n
20	RW/1S	1'h0	**Pin Mux Select Lock 4 (PMUX_SEL4)** 1b: Lock Pmux Select	pwr_rst_n
19	RW/1S	1'h0	**Pin Mux Select Lock 3 (PMUX_SEL3)** 1b: Lock Pmux Select	pwr_rst_n
18	RW/1S	1'h0	**Pin Mux Select Lock 2 (PMUX_SEL2)** 1b: Lock Pmux Select	pwr_rst_n
17	RW/1S	1'h0	**Pin Mux Select Lock 1 (PMUX_SEL1)** 1b: Lock Pmux Select	pwr_rst_n
16	RW/1S	1'h0	**Pin Mux Select Lock 0 (PMUX_SEL0)** 1b: Lock Pmux Select	pwr_rst_n
15	RW/1S	1'h0	**Pin Mux Select Lock 15 (PMUX_SELY15)** 1b: Lock Pmux Select	pwr_rst_n
14	RW/1S	1'h0	**Pin Mux Select Lock 14 (PMUX_SELY14)** 1b: Lock Pmux Select	pwr_rst_n
13	RW/1S	1'h0	**Pin Mux Select Lock 13 (PMUX_SELY13)** 1b: Lock Pmux Select	pwr_rst_n
12	RW/1S	1'h0	**Pin Mux Select Lock 12 (PMUX_SELY12)** 1b: Lock Pmux Select	pwr_rst_n
11	RW/1S	1'h0	**Pin Mux Select Lock 11 (PMUX_SELY11)** 1b: Lock Pmux Select	pwr_rst_n
10	RW/1S	1'h0	**Pin Mux Select Lock 10 (PMUX_SELY10)** 1b: Lock Pmux Select	pwr_rst_n
9	RW/1S	1'h0	**Pin Mux Select Lock 9 (PMUX_SELY9)** 1b: Lock Pmux Select	pwr_rst_n
8	RW/1S	1'h0	**Pin Mux Select Lock 8 (PMUX_SELY8)** 1b: Lock Pmux Select	pwr_rst_n
7	RW/1S	1'h0	**Pin Mux Select Lock 7 (PMUX_SELY7)** 1b: Lock Pmux Select	pwr_rst_n
6	RW/1S	1'h0	**Pin Mux Select Lock 6 (PMUX_SELY6)** 1b: Lock Pmux Select	pwr_rst_n
5	RW/1S	1'h0	**Pin Mux Select Lock 5 (PMUX_SELY5)** 1b: Lock Pmux Select	pwr_rst_n
4	RW/1S	1'h0	**Pin Mux Select Lock 4 (PMUX_SELY4)** 1b: Lock Pmux Select	pwr_rst_n

Bits	Access Type	Default	Description	ResetSignal
3	RW/1S	1'h0	**Pin Mux Select Lock 3 (PMUX_SELY3)** 1b: Lock Pmux Select	pwr_rst_n
2	RW/1S	1'h0	**Pin Mux Select Lock 2 (PMUX_SELY2)** 1b: Lock Pmux Select	pwr_rst_n
1	RW/1S	1'h0	**Pin Mux Select Lock 1 (PMUX_SELY1)** 1b: Lock Pmux Select	pwr_rst_n
0	RW/1S	1'h0	**Pin Mux Select Lock 0 (PMUX_SELY0)** 1b: Lock Pmux Select	pwr_rst_n

24.3.1.74 Pin Mux Slew Rate Lock (PMUX_IN_EN_LOCK)

MEM Offset (00000000)	960h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description	ResetSignal
31:1	RO	31'h0000_0000	**RSVD (RSVD)** Reserved	
0	RW/1S	1'h0	**Pin Mux Input enable 0 Lock (PMUX_IN0_EN_LOCK)** 1b: Lock Input Enable	pwr_rst_n

24.3.1.75 Identification Register (ID)

MEM Offset (00000000)	1000h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0010h

Bits	Access Type	Default	Description
31:8	RO	24'h000000	**RSVD (RSVD)** Reserved
7:0	RO	8'h10	**Major Revision ID (ID)** Identifies SoC revision. 10h : Revision 1.0

24.3.1.76 Revision Register (REV)

MEM Offset (00000000)	1004h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0000h

Bits	Access Type	Default	Description
31:8	RO	24'h000000	**RSVD (RSVD)** Reserved
7:0	RO	8'h00	**Minor Revision ID (REV)** Read-Only Revision register at fixed location in Memory map.

24.3.1.77 Flash Size Register (FS)

MEM Offset (00000000)	1008h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0024h

Bits	Access Type	Default	Description
31:16	RO	16'h0000	**RSVD (RSVD)** Reserved
15:0	RO	16'h0024	**Flash Size Register (FS)** Indicates Flash Size in KB.

24.3.1.78 RAM Size Register (RS)

MEM Offset (00000000)	100Ch
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0008h

Bits	Access Type	Default	Description
31:16	RO	16'h0000	**RSVD (RSVD)** Reserved
15:0	RO	16'h0008	**RAM Size Register (RS)** Indicates RAM Size in KB.

24.3.1.79 Code OTP Size Register (COTPS)

MEM Offset (00000000)	1010h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0008h

Bits	Access Type	Default	Description
31:16	RO	16'h0000	**RSVD (RSVD)** Reserved
15:0	RO	16'h0008	**Code OTP Size Register (COTPS)** Indicates Code OTP Size in KB.

24.3.1.80 Data OTP Size Register (DOTPS)

MEM Offset (00000000)	1014h
Security_PolicyGroup	
IntelRsvd	False
Size	32 bits
Default	0000_0004h

Bits	Access Type	Default	Description
31:16	RO	16'h0000	**RSVD (RSVD)** Reserved
15:0	RO	16'h0004	**Data OTP Size Register (DOTPS)** Indicates Data OTP Size in KB.

§

25 AON Counters

The SoC supports 2 Always On (AON) counters.

The first counter is an always on free running counter running off the 32kHz RTC clock. The second is a periodic counter which allows a timer value to be loaded, and an interrupt to fire when the timer expires.

25.1 Features

25.1.1 AON Counter

The following is a list of AON counters features:

- A free running up counter running off the 32,768Hz clock
- Can be enabled/disabled via software
- Can be used as a general purpose timer
- Can be used by SW for time-stamping samples received from the sensors/ADC

25.1.2 AON Periodic Timer

The following is a list of AON periodic timer features:

- Periodic counter running off the 32,768Hz clock
- AON Periodic Timer continuously decrements from a configured value
- AON Periodic Timer expires when the counter reaches 0
- When the AON Periodic Timer expires it reloads a configured value and decrements
- The AON Periodic Timer is disabled by loading a value of 0, it is enabled by loading a non 0 value
- Generates an alarm when the timer expires

The AON Control & Status registers are described in the previous chapter on the System Control Subsystem. The AON Periodic Timer configuration comes from the system clock domain – the AON Periodic Timer is clocked using the RTC clock, so it will take a number of core clocks for the AON Periodic Timer configuration to propagate into the RTC clock domain. To program the AON Periodic Timer (i.e. clear an alarm via the AONPT_CTRL.AONPT_CLR register bit or reset the counter value via the AONPT_CTRL.AONPT_RST) the relevant control bit must be set by software and then polled until it is cleared by hardware. At this point the desired configuration has taken effect.

AON Periodic timer runs on RTC Clock. Whenever power is cycled, RTC Oscillator takes time to lock and its lock time (< 2ms) is typically higher than Hybrid Oscillator (< 2 us) that provides the system clock. Thus CPU may start to execute much earlier, before RTC oscillator has attained lock. For AON Periodic timer to be started through AONPT_RST=1, RTC Clock has to be running already.

Before setting AONPT_RST register to 1, check if RTC OSC has attained lock and is running. This is achieved by checking for non-zero value of AON Counter, which free-runs on RTC clock if enabled.

The counter can the reset to the value contained in (AONPT_CFG) by writing to 1 to (AONPT_RST). (AONPT_RST) is self-clearing however due to clock domain crossing of register value from the slow to the fast domain, it needs to be polled to ensure the reset has occurred.

1. Check if RTC Clock is running by polling for non-zero value on AON Counter. Wait till AON Counter is non-zero (assumption is that AON Counter is already enabled by AONC_CFG. **AONC_CNT_EN = 1**).
2. Write the timeout value to (**AONPT_CFG**) to stop the counter
3. Write 1 to (**AONPT_RST**) to apply the counter value.
4. Poll (**AONPT_RST**) till it clears to 0 to ensure the operation has complete.
5. At this time AONPT Counter in RTC Clock domain has started running.

AON Periodic Timer is running in RTC Clock domain. It will take 1 RTC clock cycle to get reflected in AONPT_CNT read data. So AONPT_CNT read data value (in system clock domain) at any time would be 1 RTC clock cycle old value of AONPT counter in RTC clock domain.

Note that step (3) can take a number of 32kHz clocks (1000's of 32MHz clocks to complete)

The sequence for clearing an interrupt should be as follows

1. Interrupt asserted
2. Write 0 to the (AONPT_CFG) to stop the counter
3. Write 1 to (AONPT_CLR) to clear the interrupt
4. Poll (AONPT_CLR) to ensure the operation has complete

Note that step (3) can take a number of 32kHz clocks (1000's of 32MHz clocks to complete)

In addition, if both AON Counter and AON Periodic timer is not used, in order to conserve power, the clocks to these block can be gated by resetting CCU_LP_CLK_CTL.CCU_AON_TMR_CNT_CLK_EN_SW register bit to 0 in SCSS.

§