

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

"Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.

8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

H8S/2140B Group

Hardware Manual

Renesas 16-Bit Single-Chip
Microcomputer

H8S Family/H8S/2100 Series

H8S/2161B HD64F2161BV

H8S/2160B HD64F2160BV

H8S/2141B HD64F2141BV

H8S/2140B HD64F2140BV

H8S/2145B HD64F2145BV

HD64F2145B

H8S/2148B HD64F2148BV

HD64F2148B

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any rights, including but not limited to, patent, copyright, trademark, or other intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs, and algorithms represents information on products at the time of publication of these materials, and is subject to change by Renesas Technology Corp. without notice due to product improvement or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss resulting from these inaccuracies or errors.
Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the use of information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a critical system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for more information considering the use of a product contained herein for any specific purposes, such as apparatuses or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce these materials, in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they may not be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

2. Treatment of Unused Input Pins

Note: Fix all unused input pins to high or low level.

Generally, the input pins of CMOS products are high-impedance input pins. If they are in their open states, intermediate levels are induced by noise in the vicinity through current flows internally, and a malfunction may occur.

3. Processing before Initialization

Note: When power is first supplied, the product's state is undefined.

The states of internal circuits are undefined until full power is supplied through the chip and a low level is input on the reset pin. During the period where the state is undefined, the register settings and the output state of each pin are also undefined. Be sure to initialize your system so that it does not malfunction because of processing while it is in an undefined state. For those products which have a reset function, reset the LSI after the power supply has been turned on.

4. Prohibition of Access to Undefined or Reserved Addresses

Note: Access to undefined or reserved addresses is prohibited.

The undefined or reserved addresses may be used to expand functions, or test data may have been allocated to these addresses. Do not access these registers; test operation is not guaranteed if they are accessed.

Rev. 3.00 Mar 21, 2006

RENESAS

The list of revisions is a summary of points that have been revised or added to earlier editions. This does not include all of the revised contents. For details, see the actual locations in the manual.

5. Contents
6. Overview
7. Description of Functional Modules
 - CPU and System-Control Modules
 - On-Chip Peripheral Modules

The configuration of the functional description of each module differs according to module. However, the generic style includes the following items:

- i) Feature
- ii) Input/Output Pin
- iii) Register Description
- iv) Operation
- v) Usage Note

When designing an application system that includes this LSI, take notes into account. Each section includes notes in relation to the descriptions given, and usage notes are given, as required, in the final part of each section.

8. List of Registers
9. Electrical Characteristics
10. Appendix
11. Index

This LSI is equipped with a data transfer controller (DTC) as a bus master, ROM, RAM, PWM timer (PWM), a 14-bit PWM timer (PWMX), a 16-bit free-running timer (FRT), timer (TMR), timer connection, a watchdog timer (WDT), a serial communication interface, a keyboard buffer controller, a host interface X-bus interface (XBS), a host interface L (LPC), an 8-bit D/A converter, a 10-bit A/D converter, and I/O ports as on-chip peripheral modules required for system configuration. An I²C bus interface (IIC) can also be included as an optional interface.

A high-functionality bus controller is also provided, enabling fast and easy connection to various kinds of memory and other kinds of memory.

A flash memory (F-ZTAT^{TM*}) version is available for this LSI's ROM. This provides a version that can be reprogrammed in no time to cope with all situations from the early stages of development to production to full-scale mass production. This is particularly applicable to applications where specifications that will most probably change.

Note: * F-ZTAT is a trademark of Renesas Technology Corp.

Target Users: This manual was written for users who will be using the H8S/2140B Group design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of the H8S/2140B Group to the target users. Refer to the H8S/2600 Series, H8S/2000 Series Programming Manual for a detailed description of the instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip
Read the manual according to the contents. This manual can be roughly categorized on the CPU, system control functions, peripheral functions and electrical characteristics.

Rev. 3.00 Mar 21, 2006

RENESAS

implemented on more than one channel.
XXX_N (XXX is the register name and N is the
number)

Bit order: The MSB is on the left and the LSB is on the right
Number notation: Binary is B'xxxx, hexadecimal is H'xxxx, decimal is D'xxxx
Signal notation: An overbar is added to a low-active signal: $\overline{\text{xxxx}}$

Related Manuals: The latest versions of all related manuals are available from our website.
Please ensure you have the latest versions of all documents you refer to.
<http://www.renesas.com/>

H8S/2140B Group manuals:

Document Title	Document ID
H8S/2140B Group Hardware Manual	This manual
H8S/2600 Series, H8S/2000 Series Programming Manual	REJ09B0010

User's manuals for development tools:

Document Title	Document ID
H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B0010
H8S, H8/300 Series Simulator/Debugger User's Manual	ADE-7B0010
H8S, H8/300 Series High-performance Embedded Workshop, High-performance Debugging Interface Tutorial	ADE-7B0010
High-performance Embedded Workshop User's Manual	ADE-7B0010

various peripheral functions

Note * added
Host Interface LPC interface*

Note: * The LPC function is not supported by H8S/2145B (5-V version).

2 • On-chip memory

Table amended

ROM	Model	ROM	RAM	Remarks
F-ZTAT Version	HD64F2161BV*	128 kbytes	4 kbytes	
	HD64F2160BV*	64 kbytes	4 kbytes	
	HD64F2141BV*	128 kbytes	4 kbytes	
	HD64F2140BV*	64 kbytes	4 kbytes	
	HD64F2145BV*	256 kbytes	8 kbytes	Under de
	HD64F2145B	256 kbytes	8 kbytes	
	HD64F2148BV*	128 kbytes	4 kbytes	
	HD64F2148B	128 kbytes	4 kbytes	

• Compact package

Table amended

(Before) 18.0 × 18.0 mm → (After) 16.0 × 16.0 mm
(Before) 16.0 × 16.0 mm → (After) 14.0 × 14.0 mm

1.2 Block Diagram 3

Figure 1.1 Internal Block Diagram of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B

Note amended

Note: * The LPC function and the \overline{WUE} pin function supported by the H8S/2148B and H8S/2145B (5-V

Figure 1.2 Internal Block Diagram of H8S/2160B and H8S/2161B 4

Figure 1.2 Internal Block Diagram of H8S/2160B and H8S/2161B

Figure 1.2 amended

(Before) ROM (Flash memory, Masked ROM) → (Flash memory)

Rev. 3.00 Mar 21, 2006



1.3.2 Pin Functions in Each Operating Mode

Table 1.1 Pin Functions of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B in Each Operating Mode

Note: * amended

Note: * The LPC function and the \overline{WUE} pin function supported by the H8S/2148B and H8S/2145B (5-V v

2.4.4 Condition-Code Register (CCR) 36

Table amended
Interrupt Mask Bit

Masks interrupts when set to 1. NMI is accepted ...

2.6.1 Table of Instructions Classified by Function 46

Table amended

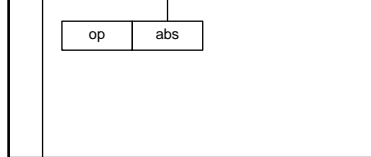
Table 2.7 Bit Manipulation Instructions (1)

Instruction	Size*	Function
BIAND	B	$C \wedge \overline{[(-\langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle)]} \rightarrow C$ Logically ANDs the carry flag with the inverse of a specified general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.
BIOR	B	$C \vee \overline{[(-\langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle)]} \rightarrow C$ Logically ORs the carry flag with the inverse of a specified general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.

2.7.5 Absolute Address—@aa:8, @aa:16, @aa:24, or @aa:32 53

Description amended

... absolute address, the upper 24 bits are all assumed to be 0.

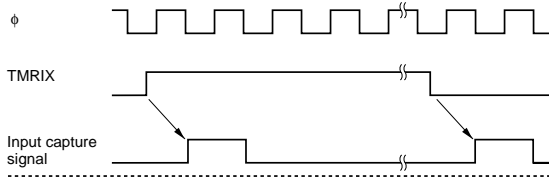
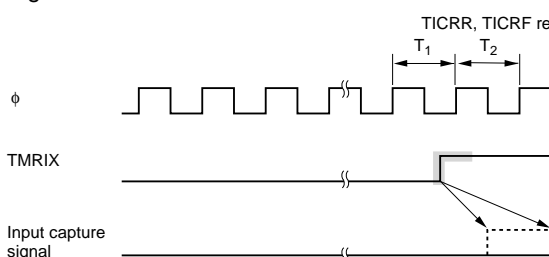


3.4 Address Map in Each Operating Mode	77, 78	Figure 3.7 and Figure 3.8 added
Figure 3.7 Address Map for H8S/2145B (1)		
Figure 3.8 Address Map for H8S/2145B (2)		
Figure 3.9 Address Map for H8S/2148B(1)	79	Figure 3.9 amended H'01FFFF H'020000 H'FFE080
4.7 Usage Note	87	Figure 4.3 amended
Figure 4.3 Operation when SP Value Is Old		H'FFEFFF
5.2 Input/Output Pins	91	Note * amended
Table 5.1 Pin Configuration		Note: * Not supported by the H8S/2148B and H8S/2148B version).
5.3.7 Keyboard Matrix Interrupt Mask Registers (KMIMRA, KMIMR) and Wake-Up Event Mask Register (WUEMRB)	98	• WUEMRB* Note * amended Note: * Not supported by the H8S/2148B and H8S/2148B version).

Rev. 3.00 Mar 21, 2006



5.6.5	DTC Activation by Interrupt	114	Description amended ... the DTCE bit of DTC's DT CER , and the DIESEL bit
7.2.8	DTC Vector Register (DTVECR)	151	Description amended ... software activation interrupt. DTVECR is initialized to H'00 at a reset and in hardware standby mode.
7.4	Location of Register Information and DTC Vector Table Table 7.1 Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs	154	Note 2 amended Note: 2. Not supported by the H8S/2148B and H8S/2148B (version).
8.1	Overview	167	Description amended ... in addition to DDR , to control the on/off ...
	Table 8.1 Port Functions of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B	171	Note * amended Note: * Not supported by the H8S/2148B and H8S/2148B (version).
8.4.4	Pin Functions	181	• P37/D15/HDB7/SERIRQ*, ... , P30/D8/HDB0/LAD Note amended Note: * Not supported by the H8S/2148B and H8S/2148B (version).

	199	<ul style="list-style-type: none"> • P80/HA0/PME*³ <p>Note 3 amended</p> <p>Note: 3. Not supported by the H8S/2148B and H8S/2148C (version).</p>
8.12.4 Pin Functions	213 to 215	<ul style="list-style-type: none"> • PB7/D7/WUE7*², PB6/D6/WUE6*², PB5/D5/WUE5*², PB4/D4/WUE4*² to PB0/D0/WUE0/HIRQ3/LSMI*⁴ <p>Notes amended</p> <p>Note: Not supported by the H8S/2148B and H8S/2148C (version).</p>
11.3.6 Timer Interrupt Enable Register (TIER)	265	<p>Table amended</p> <p>Initial Value (Before) 0 → (After) 1</p>
12.7 Input Capture Operation	307	<p>Figure 12.11 amended</p>  <p>Figure 12.11 Timing of Input Capture Operation</p>
Figure 12.12 Timing of Input Capture Signal (Input capture signal is input during TICRR and TICRF read)	307	<p>Figure 12.12 amended</p>  <p>Figure 12.12 Timing of Input Capture Signal (Input capture signal is input during TICRR and TICRF read)</p>

Rev. 3.00 Mar 21, 2006

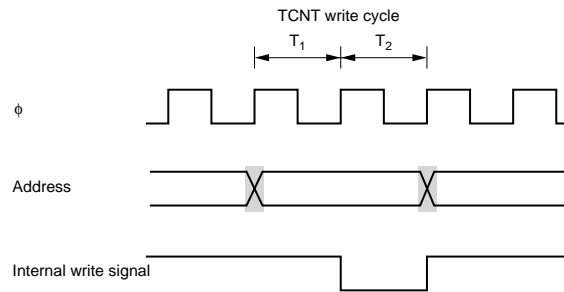
RENESAS

14.6.2 Conflict between Timer Counter (TCNT) Write and Increment

357

Figure 14.7 amended

Figure 14.7 Conflict between TCNT Write and Increment



15.1 Features

360

Figure legend amended

Figure 15.1 Block Diagram of SCI

(Before) SCMR: Smart card mode register → (After) Serial interface mode register

16.3.5 I²C Bus Control Register (ICCR)

424

Table amended

Bit	Bit Name	Initial Value	R/W	Description
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select
				00: Slave receive mode
				01: Slave transmit mode
				10: Master receive mode
				11: Master transmit mode
				Both these bits will be cleared by hardware in case of a bus contention in master mode with I ² C bus format. In slave receive mode with I ² C bus format, the R/W bit in the first frame immediately after the condition sets these bits in receive mode or automatically by hardware.
				Modification of the TRS bit during transfer is not allowed. The changeover to receive mode is completed after the transfer is completed, and the changeover to transmit mode is completed after the completion of the transfer.

16.4.4 Master Receive Operation

Figure 16.12 Example of Operation Timing in Master Receive Mode (MLS = WAIT = 0, HNDS = 1)

Figure 16.12 amended

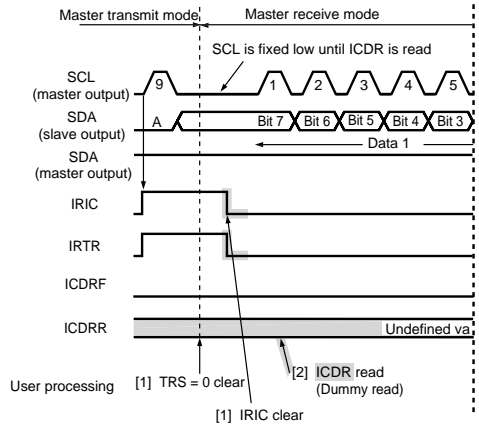


Figure 16.13 Example of Stop Condition Issuance Operation Timing in Master Receive Mode (MLS = WAIT = 0, HNDS = 1)

Figure 16.13 amended

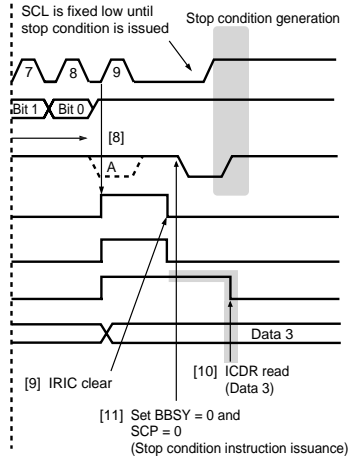
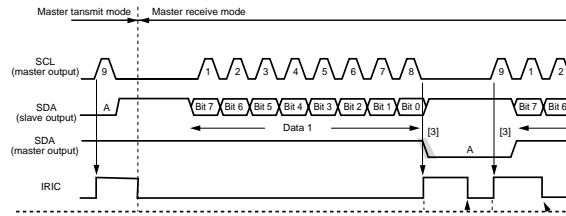


Figure 16.16 Example of Master Receive Mode Operation Timing (MLS = ACKB = 0, WAIT = 1)

Figure 16.16 amended



16.4.5 Slave Receive Operation 464

Figure title amended

Figure 16.22 Example of Slave Receive Mode Operation Timing (1) (MLS = ACKB = 0, HNDS = 0)

Figure 16.23 Example of Slave Receive Mode Operation Timing (2) (MLS = ACKB = 0, HNDS = 0)

Figure title amended

16.6 Usage Notes 483

10. Notes on WAIT Function

Description added

Figure 16.35 ICDR Read and ICCR Access Timing in Slave Transmit Mode

Figure 16.35 amended

R/W

487,
488

14. Notes on Arbitration Lost in Master Mode

Description added

19.4.4 Host Interface Shutdown Function (LPCPD)

569

Table 19.5 amended

Table 19.5 Scope of Host Interface Pin Shutdown

Abbreviation	Port	Scope of Shutdown	I/O	Notes
CLKRUN	P82	O	Input	Hi-Z
LPCPD	P83	×	Input	Needed to clear shutdown

Section 22 RAM 601

Masked ROM version deleted

Product Classification		RAM Capacitance	RAM Address
Flash memory version	H8S/2161B	4 kbytes	H'E080-H'FFFF, H
	H8S/2160B	4 kbytes	H'E080-H'FFFF, H
	H8S/2141B	4 kbytes	H'E080-H'FFFF, H
	H8S/2140B	4 kbytes	H'E080-H'FFFF, H
	H8S/2145B	8 kbytes	H'D080-H'FFFF, H
	H8S/2148B	4 kbytes	H'E080-H'FFFF, H

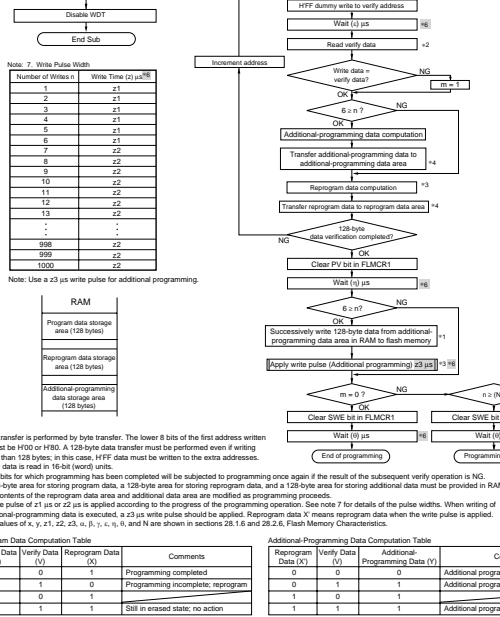
Section 23 ROM 603

Description amended

(Before) This LSI has an on-chip ROM (flash memory masked ROM), ... → (After) This LSI has an on-chip memory. ...

Rev. 3.00 Mar 21, 2006





Section of "Masked ROM" deleted

Section 24 Clock Pulse Generator 633

Figure 24.1 amended (Before) φ2 to f32 → (After) φ2 to φ32

Figure 24.1 Block Diagram of Clock Pulse Generator

24.5 Subclock Input Circuit 639

Description of "When Subclock Is Not Needed" and "Subclock Usage" added

25.1.1 Standby Control Register (SBYCR) 643

Table amended ... SCK2 to SCK0 must be cleared to B'000.



Order)			version).
26.2	Register Bits	679	Note 5 amended Note: 5. Not supported by the H8S/2148B and H8S version).
26.3	Register States in Each Operating Mode	688	Note 2 amended Note: 2. Not supported by the H8S/2148B and H8S version).
26.4	Register Select Conditions	690	Table amended H8S/2160B, H8S/2161B Register Select Condition (Before) — → (After) No condition
27.1.1	Absolute Maximum Ratings Table 27.1 Absolute Maximum Ratings	701	Table 27.1 amended item (Ports C to G are added in the H8S/2160B and H8S input voltage (P97, P86, P52, P42)
27.1.2	DC Characteristics Table 27.2 DC Characteristics (1)	703	Table 27.2 amended Item P97, P86, P52, P42 (Ports C to G are added in the H8S/2160B and H8S

Rev. 3.00 Mar 21, 2006 p

RENESAS

are guaranteed after rewriting (Guarantee range is 1 minimum value).

9. Reference value for 25°C (as a guideline, rewriting normally function up to this value).

10. Data retention characteristic when rewriting is pe within the specification range, including the minimum

27.2.2 DC Characteristics

736

Table 27.17 (5) amended

Table 27.17 DC Characteristics (5)

	Item	Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60 (KWUL (1) = 00) ^{92,96}	V_T^-	$V_{CC} \times 0.2$	—	—	V
	KIN15 to KIN8 ^{97,98} , IRQ2 to IRQ0 ⁹³ , IRQ5 to IRQ3	V_T^+	—	—	$V_{CC} \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	
				$V_{CC} \times 0.05$	—	

27.2.3 AC Characteristics

752

Table 27.23 (1) amended

Table 27.23 Timing of On-Chip Peripheral Modules (1)

(Before) t_{csyc} → (After) t_{soyc}

27.2.7 Usage Notes

761

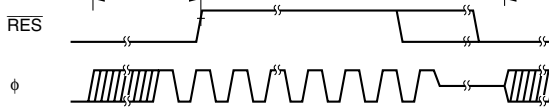
Figure 27.5 amended

Figure 27.5 Connection of VCL Capacitor

< Product with internal step-down function >

HD64F2145B

HD64F2148B



Appendix B Product 779 Table amended
Codes

Product Type	Product Code	Mark Code	Package (Packag
H8S/2145B Flash memory version (3-V version)	HD64F2145BV	F2145BVFA10	100-pin
		F2145BVTE10	100-pin
Flash memory version (5-V version)	HD64F2145B	F2145BFA20	100-pin
		F2145BTE20	100-pin

Appendix C Package 780 Figure replaced
Dimensions

Figure C.1 Package Dimensions (FP-100B)

Figure C.2 Package 781 Figure replaced
Dimensions (TFP-100B)

Figure C.3 Package 782 Figure replaced
Dimensions (TFP-144)

Rev. 3.00 Mar 21, 2006 p

RENESAS

1.3.2	Pin Functions in Each Operating Mode
1.3.3	Pin Functions
Section 2 CPU	
2.1	Features
2.1.1	Differences between H8S/2600 CPU and H8S/2000 CPU
2.1.2	Differences from H8/300 CPU
2.1.3	Differences from H8/300H CPU
2.2	CPU Operating Modes
2.2.1	Normal Mode
2.2.2	Advanced Mode
2.3	Address Space
2.4	Register Configuration
2.4.1	General Registers
2.4.2	Program Counter (PC)
2.4.3	Extended Control Register (EXR)
2.4.4	Condition-Code Register (CCR)
2.4.5	Initial Register Values
2.5	Data Formats
2.5.1	General Register Data Formats
2.5.2	Memory Data Formats
2.6	Instruction Set
2.6.1	Table of Instructions Classified by Function
2.6.2	Basic Instruction Formats
2.7	Addressing Modes and Effective Address Calculation
2.7.1	Register Direct—Rn
2.7.2	Register Indirect—@ERn
2.7.3	Register Indirect with Displacement—@(d:16, ERn) or @(d:32, ERn)
2.7.4	Register Indirect with Post-Increment or Pre-Decrement—@ERn+ or @ERn-
2.7.5	Absolute Address—@aa:8, @aa:16, @aa:24, or @aa:32
2.7.6	Immediate—#xx:8, #xx:16, or #xx:32
2.7.7	Program-Counter Relative—@(d:8, PC) or @(d:16, PC)

Rev. 3.00 Mar 21, 2006 p

RENESAS

Section 3	MCU Operating Modes
3.1	MCU Operating Mode Selection
3.2	Register Descriptions
3.2.1	Mode Control Register (MDCR)
3.2.2	System Control Register (SYSCR)
3.2.3	Serial Timer Control Register (STCR)
3.3	Operating Mode Descriptions
3.3.1	Mode 1
3.3.2	Mode 2
3.3.3	Mode 3
3.3.4	Pin Functions in Each Operating Mode
3.4	Address Map in Each Operating Mode
Section 4	Exception Handling
4.1	Exception Handling Types and Priority
4.2	Exception Sources and Exception Vector Table
4.3	Reset.....
4.3.1	Reset Exception Handling.....
4.3.2	Interrupts after Reset.....
4.3.3	On-Chip Peripheral Modules after Reset Is Cancelled
4.4	Interrupt Exception Handling.....
4.5	Trap Instruction Exception Handling.....
4.6	Stack Status after Exception Handling.....
4.7	Usage Note.....
Section 5	Interrupt Controller
5.1	Features.....
5.2	Input/Output Pins
5.3	Register Descriptions
5.3.1	Interrupt Control Registers A to C (ICRA to ICRC).....
5.3.2	Address Break Control Register (ABRKCR).....
5.3.3	Break Address Registers A to C (BARA to BARC).....

5.6	Interrupt Control Modes and Interrupt Operation
5.6.1	Interrupt Control Mode 0
5.6.2	Interrupt Control Mode 1
5.6.3	Interrupt Exception Handling Sequence
5.6.4	Interrupt Response Times
5.6.5	DTC Activation by Interrupt.....
5.7	Address Break
5.7.1	Features
5.7.2	Block Diagram
5.7.3	Operation
5.7.4	Usage Notes
5.8	Usage Notes
5.8.1	Conflict between Interrupt Generation and Disabling
5.8.2	Instructions that Disable Interrupts
5.8.3	Interrupts during Execution of EEPMOV Instruction.....
5.8.4	Setting on Product Incorporating DTC
5.8.5	IRQ Status Register (ISR).....
Section 6 Bus Controller (BSC)	
6.1	Features
6.2	Input/Output Pins
6.3	Register Descriptions
6.3.1	Bus Control Register (BCR)
6.3.2	Wait State Control Register (WSCR)
6.4	Bus Control
6.4.1	Bus Specifications.....
6.4.2	Advanced Mode.....
6.4.3	Normal Mode.....
6.4.4	I/O Select Signals.....
6.5	Basic Bus Interface
6.5.1	Data Size and Data Alignment.....
6.5.2	Valid Strokes.....

Rev. 3.00 Mar 21, 2006 p



Section 7	Data Transfer Controller (DTC).....
7.1	Features.....
7.2	Register Descriptions.....
7.2.1	DTC Mode Register A (MRA).....
7.2.2	DTC Mode Register B (MRB).....
7.2.3	DTC Source Address Register (SAR).....
7.2.4	DTC Destination Address Register (DAR).....
7.2.5	DTC Transfer Count Register A (CRA).....
7.2.6	DTC Transfer Count Register B (CRB).....
7.2.7	DTC Enable Registers (DTCER).....
7.2.8	DTC Vector Register (DTVECR).....
7.3	Activation Sources.....
7.4	Location of Register Information and DTC Vector Table.....
7.5	Operation.....
7.5.1	Normal Mode.....
7.5.2	Repeat Mode.....
7.5.3	Block Transfer Mode.....
7.5.4	Chain Transfer.....
7.5.5	Interrupts.....
7.5.6	Operation Timing.....
7.5.7	Number of DTC Execution States.....
7.6	Procedures for Using DTC.....
7.6.1	Activation by Interrupt.....
7.6.2	Activation by Software.....
7.7	Examples of Use of DTC.....
7.7.1	Normal Mode.....
7.7.2	Software Activation.....
7.8	Usage Notes.....
7.8.1	Module Stop Mode Setting.....
7.8.2	On-Chip RAM.....
7.8.3	DTCE Bit Setting.....

	8.2.4	Pin Functions
	8.2.5	Port 1 Input Pull-Up MOS
8.3	Port 2	
	8.3.1	Port 2 Data Direction Register (P2DDR)
	8.3.2	Port 2 Data Register (P2DR)
	8.3.3	Port 2 Pull-Up MOS Control Register (P2PCR)
	8.3.4	Pin Functions
	8.3.5	Port 2 Input Pull-Up MOS
8.4	Port 3	
	8.4.1	Port 3 Data Direction Register (P3DDR)
	8.4.2	Port 3 Data Register (P3DR)
	8.4.3	Port 3 Pull-Up MOS Control Register (P3PCR)
	8.4.4	Pin Functions
	8.4.5	Port 3 Input Pull-Up MOS
8.5	Port 4	
	8.5.1	Port 4 Data Direction Register (P4DDR)
	8.5.2	Port 4 Data Register (P4DR)
	8.5.3	Pin Functions
8.6	Port 5	
	8.6.1	Port 5 Data Direction Register (P5DDR)
	8.6.2	Port 5 Data Register (P5DR)
	8.6.3	Pin Functions
8.7	Port 6	
	8.7.1	Port 6 Data Direction Register (P6DDR)
	8.7.2	Port 6 Data Register (P6DR)
	8.7.3	Port 6 Pull-Up MOS Control Register (KMPCR)
	8.7.4	Pin Functions
	8.7.5	Port 6 Input Pull-Up MOS
8.8	Port 7	
	8.8.1	Port 7 Input Data Register (P7PIN)
	8.8.2	Pin Functions

8.11.1	Port A Data Direction Register (PADDDR)
8.11.2	Port A Output Data Register (PAODR)
8.11.3	Port A Input Data Register (PAPIN).....
8.11.4	Pin Functions
8.11.5	Port A Input Pull-Up MOS
8.12	Port B
8.12.1	Port B Data Direction Register (PBDDR).....
8.12.2	Port B Output Data Register (PBODR)
8.12.3	Port B Input Data Register (PBPIN)
8.12.4	Pin Functions
8.12.5	Port B Input Pull-Up MOS
8.13	Additional Overview for H8S/2160B and H8S/2161B
8.14	Ports C, D.....
8.14.1	Port C and Port D Data Direction Registers (PCDDR, PDDDR)
8.14.2	Port C and Port D Output Data Registers (PCODR, PDODR)
8.14.3	Port C and Port D Input Data Registers (PCPIN, PDPIN).....
8.14.4	Port C and Port D Nch-OD Control Register (PCNOCR, PDNOCR)
8.14.5	Pin Functions
8.14.6	Input Pull-Up MOS in Ports C and D
8.15	Ports E, F.....
8.15.1	Port E and Port F Data Direction Registers (PEDDR, PFDDR)
8.15.2	Port E and Port F Output Data Registers (PEODR, PFODR)
8.15.3	Port E and Port F Input Data Registers (PEPIN, PFPIN).....
8.15.4	Port E and Port F Nch-OD Control Register (PENOCR, PFNOCR)
8.15.5	Pin Functions
8.15.6	Input Pull-Up MOS in Ports E and F
8.16	Port G.....
8.16.1	Port G Data Direction Register (PGDDR)
8.16.2	Port G Output Data Register (PGODR)
8.16.3	Port G Input Data Register (PGPIN).....
8.16.4	Port G Nch-OD Control Register (PGNOCR)
8.16.5	Pin Functions

9.4	Operation
9.5	Usage Note.....
9.5.1	Module Stop Mode Setting
Section 10 14-Bit PWM Timer (PWMX).....	
10.1	Features
10.2	Input/Output Pins
10.3	Register Descriptions
10.3.1	PWM (D/A) Counters H and L (DACNTH, DACNTL).....
10.3.2	PWM (D/A) Data Registers A and B (DADRA, DADRB)
10.3.3	PWM (D/A) Control Register (DACR)
10.4	Bus Master Interface
10.5	Operation
10.6	Usage Note.....
10.6.1	Module Stop Mode Setting
Section 11 16-Bit Free-Running Timer (FRT).....	
11.1	Features
11.2	Input/Output Pins
11.3	Register Descriptions
11.3.1	Free-Running Counter (FRC)
11.3.2	Output Compare Registers A and B (OCRA, OCRB)
11.3.3	Input Capture Registers A to D (ICRA to ICRD)
11.3.4	Output Compare Registers AR and AF (OCRAR, OCRAF)
11.3.5	Output Compare Register DM (OCRDM).....
11.3.6	Timer Interrupt Enable Register (TIER).....
11.3.7	Timer Control/Status Register (TCSR).....
11.3.8	Timer Control Register (TCR).....
11.3.9	Timer Output Compare Control Register (TOCR)
11.4	Operation
11.4.1	Pulse Output.....
11.5	Operation Timing.....

Rev. 3.00 Mar 21, 2006 pa



	11.5.10 Mask Signal Generation Timing
11.6	Interrupt Sources
11.7	Usage Notes
	11.7.1 Conflict between FRC Write and Clear
	11.7.2 Conflict between FRC Write and Increment.....
	11.7.3 Conflict between OCR Write and Compare-Match
	11.7.4 Switching of Internal Clock and FRC Operation
	11.7.5 Module Stop Mode Setting
Section 12	8-Bit Timer (TMR).....
12.1	Features
12.2	Input/Output Pins
12.3	Register Descriptions
	12.3.1 Timer Counter (TCNT).....
	12.3.2 Time Constant Register A (TCORA).....
	12.3.3 Time Constant Register B (TCORB)
	12.3.4 Timer Control Register (TCR).....
	12.3.5 Timer Control/Status Register (TCSR).....
	12.3.6 Input Capture Register (TICR)
	12.3.7 Time Constant Register (TCORC).....
	12.3.8 Input Capture Registers R and F (TICRR, TICRF).....
	12.3.9 Timer Input Select Register (TISR)
12.4	Operation
	12.4.1 Pulse Output.....
12.5	Operation Timing.....
	12.5.1 TCNT Count Timing.....
	12.5.2 Timing of CMFA and CMFB Setting at Compare-Match
	12.5.3 Timing of Timer Output at Compare-Match
	12.5.4 Timing of Counter Clear at Compare-Match
	12.5.5 TCNT External Reset Timing
	12.5.6 Timing of Overflow Flag (OVF) Setting
12.6	Operation with Cascaded Connection

	12.9.5	Switching of Internal Clocks and TCNT Operation.....	
	12.9.6	Mode Setting with Cascaded Connection	
	12.9.7	Module Stop Mode Setting	
Section 13 Timer Connection.....			
13.1		Features	
13.2		Input/Output Pins	
13.3		Register Descriptions	
	13.3.1	Timer Connection Register I (TCONRI)	
	13.3.2	Timer Connection Register O (TCONRO)	
	13.3.3	Timer Connection Register S (TCONRS).....	
	13.3.4	Edge Sense Register (SEDGR)	
13.4		Operation	
	13.4.1	PWM Decoding (PDC Signal Generation)	
	13.4.2	Clamp Waveform Generation (CL1/CL2/CL3 Signal Generation)	
	13.4.3	Measurement of 8-Bit Timer Divided Waveform Period	
	13.4.4	2fH Modification of IHI Signal	
	13.4.5	IVI Signal Fall Modification and IHI Synchronization	
	13.4.6	Internal Synchronization Signal Generation (IHG/IVG/CL4 Signal Generation)	
	13.4.7	HSYNCO Output.....	
	13.4.8	VSYNCO Output.....	
	13.4.9	CBLANK Output.....	
13.5		Usage Note.....	
	13.5.1	Module Stop Mode Setting	
Section 14 Watchdog Timer (WDT).....			
14.1		Features	
14.2		Input/Output Pins	
14.3		Register Descriptions	
	14.3.1	Timer Counter (TCNT).....	
	14.3.2	Timer Control/Status Register (TCSR).....	

Rev. 3.00 Mar 21, 2006 pa



14.6.4	Switching between Watchdog Timer Mode and Interval Timer Mode.....
14.6.5	System Reset by $\overline{\text{RESO}}$ Signal.....
14.6.6	Counter Values during Transitions between High-Speed, Sub-Active, and Watch Modes

Section 15 Serial Communication Interface (SCI and IrDA).....

15.1	Features.....
15.2	Input/Output Pins
15.3	Register Descriptions
15.3.1	Receive Shift Register (RSR)
15.3.2	Receive Data Register (RDR).....
15.3.3	Transmit Data Register (TDR).....
15.3.4	Transmit Shift Register (TSR)
15.3.5	Serial Mode Register (SMR).....
15.3.6	Serial Control Register (SCR).....
15.3.7	Serial Status Register (SSR)
15.3.8	Serial Interface Mode Register (SCMR).....
15.3.9	Bit Rate Register (BRR)
15.3.10	Keyboard Comparator Control Register (KBCOMP).....
15.4	Operation in Asynchronous Mode
15.4.1	Data Transfer Format.....
15.4.2	Receive Data Sampling Timing and Reception Margin in Asynchronous Mode.....
15.4.3	Clock.....
15.4.4	SCI Initialization (Asynchronous Mode).....
15.4.5	Data Transmission (Asynchronous Mode).....
15.4.6	Serial Data Reception (Asynchronous Mode).....
15.5	Multiprocessor Communication Function.....
15.5.1	Multiprocessor Serial Data Transmission
15.5.2	Multiprocessor Serial Data Reception
15.6	Operation in Clocked Synchronous Mode
15.6.1	Clock.....

15.9.2	Break Detection and Processing
15.9.3	Mark State and Break Detection
15.9.4	Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only)
15.9.5	Relation between Writing to TDR and TDRE Flag
15.9.6	Restrictions on Using DTC
15.9.7	SCI Operations during Mode Transitions
15.9.8	Notes on Switching from SCK Pins to Port Pins
Section 16 I²C Bus Interface (IIC) (Optional)	
16.1	Features
16.2	Input/Output Pins
16.3	Register Descriptions
16.3.1	I ² C Bus Data Register (ICDR)
16.3.2	Slave Address Register (SAR)
16.3.3	Second Slave Address Register (SARX)
16.3.4	I ² C Bus Mode Register (ICMR)
16.3.5	I ² C Bus Control Register (ICCR)
16.3.6	I ² C Bus Status Register (ICSR)
16.3.7	DDC Switch Register (DDCSWR)
16.3.8	I ² C Bus Extended Control Register (ICXR)
16.4	Operation
16.4.1	I ² C Bus Data Format
16.4.2	Initialization
16.4.3	Master Transmit Operation
16.4.4	Master Receive Operation
16.4.5	Slave Receive Operation
16.4.6	Slave Transmit Operation
16.4.7	IRIC Setting Timing and SCL Control
16.4.8	Automatic Switching from Formatless Mode to I ² C Bus Format
16.4.9	Operation Using DTC
16.4.10	Noise Canceler

Rev. 3.00 Mar 21, 2006 pa



17.3.1	Keyboard Control Register H (KBCRH)
17.3.2	Keyboard Control Register L (KBCRL)
17.3.3	Keyboard Data Buffer Register (KBBR)
17.4	Operation
17.4.1	Receive Operation
17.4.2	Transmit Operation
17.4.3	Receive Abort
17.4.4	KCLKI and KDI Read Timing
17.4.5	KCLKO and KDO Write Timing
17.4.6	KBF Setting Timing and KCLK Control
17.4.7	Receive Timing
17.4.8	KCLK Fall Interrupt Operation
17.5	Usage Notes
17.5.1	KBIOE Setting and KCLK Falling Edge Detection
17.5.2	Module Stop Mode Setting

Section 18 Host Interface X-Bus Interface (XBS).....

18.1	Features
18.2	Input/Output Pins
18.3	Register Descriptions
18.3.1	System Control Register 2 (SYSCR2)
18.3.2	Host Interface Control Register (HICR) Host Interface Control Register (HICR2)
18.3.3	Input Data Register (IDR)
18.3.4	Output Data Register 1 (ODR)
18.3.5	Status Register (STR)
18.4	Operation
18.4.1	Host Interface Activation
18.4.2	Control States
18.4.3	A20 Gate
18.4.4	Host Interface Pin Shutdown Function
18.5	Interrupt Sources

19.3	Register Descriptions
19.3.1	Host Interface Control Registers 0 and 1 (HICR0, HICR1).....
19.3.2	Host Interface Control Registers 2 and 3 (HICR2, HICR3).....
19.3.3	LPC Channel 3 Address Register (LADR3)
19.3.4	Input Data Registers 1 to 3 (IDR1 to IDR3)
19.3.5	Output Data Registers 1 to 3 (ODR1 to ODR3).....
19.3.6	Bidirectional Data Registers 0 to 15 (TWR0 to TWR15)
19.3.7	Status Registers 1 to 3 (STR1 to STR3).....
19.3.8	SERIRQ Control Registers 0 and 1 (SIRQCR0, SIRQCR1)
19.3.9	Host Interface Select Register (HISEL).....
19.4	Operation
19.4.1	Host Interface Activation.....
19.4.2	LPC I/O Cycles
19.4.3	A20 Gate
19.4.4	Host Interface Shutdown Function (LPCPD).....
19.4.5	Host Interface Serialized Interrupt Operation (SERIRQ)
19.4.6	Host Interface Clock Start Request (CLKRUN)
19.5	Interrupt Sources.....
19.5.1	IBFI1 to IBFI3, and ERRI.....
19.5.2	SMI, HIRQ1, HIRQ6, HIRQ9 to HIRQ12
19.6	Usage Notes
19.6.1	Module Stop Mode Setting
19.6.2	Notes on Using Host Interface.....
Section 20 D/A Converter	
20.1	Features
20.2	Input/Output Pins
20.3	Register Descriptions
20.3.1	D/A Data Registers 0 and 1 (DADR0, DADR1)
20.3.2	D/A Control Register (DACR)
20.4	Operation
20.5	Usage Note.....

21.3.4	Keyboard Comparator Control Register (KBCOMP)
21.4	Operation
21.4.1	Single Mode
21.4.2	Scan Mode
21.4.3	Input Sampling and A/D Conversion Time
21.4.4	External Trigger Input Timing
21.5	Interrupt Sources
21.6	A/D Conversion Accuracy Definitions
21.7	Usage Notes
21.7.1	Permissible Signal Source Impedance
21.7.2	Influences on Absolute Accuracy
21.7.3	Setting Range of Analog Power Supply and Other Pins
21.7.4	Notes on Board Design
21.7.5	Notes on Noise Countermeasures
21.7.6	Module Stop Mode Setting

Section 22 RAM

Section 23 ROM

23.1	Features
23.2	Mode Transitions
23.3	Block Configuration
23.3.1	Block Configuration of 64-Kbyte Flash Memory
23.3.2	Block Configuration of 128-Kbyte Flash Memory
23.3.3	Block Configuration of 256-Kbyte Flash Memory
23.4	Input/Output Pins
23.5	Register Descriptions
23.5.1	Flash Memory Control Register 1 (FLMCR1)
23.5.2	Flash Memory Control Register 2 (FLMCR2)
23.5.3	Erase Block Registers 1 and 2 (EBR1, EBR2)
23.6	Operating Modes
23.7	On-Board Programming Modes



23.10	Interruptions during Flash Memory Programming/Erasing
23.11	Programmer Mode
23.12	Usage Notes
Section 24 Clock Pulse Generator	
24.1	Oscillator.....
24.1.1	Connecting Crystal Resonator
24.1.2	External Clock Input Method.....
24.2	Duty Correction Circuit
24.3	Medium-Speed Clock Divider
24.4	Bus Master Clock Select Circuit
24.5	Subclock Input Circuit
24.6	Subclock Waveform Forming Circuit.....
24.7	Clock Select Circuit
24.8	Processing for X1 and X2 Pins
24.9	Usage Notes
24.9.1	Note on Resonator
24.9.2	Notes on Board Design
Section 25 Power-Down Modes.....	
25.1	Register Descriptions
25.1.1	Standby Control Register (SBYCR)
25.1.2	Low-Power Control Register (LPWRCR)
25.1.3	Module Stop Control Registers H and L (MSTPCRH, MSTPCRL).....
25.2	Mode Transitions and LSI States
25.3	Medium-Speed Mode.....
25.4	Sleep Mode
25.5	Software Standby Mode.....
25.6	Hardware Standby Mode
25.7	Watch Mode.....
25.8	Subsleep Mode.....
25.9	Subactive Mode

Rev. 3.00 Mar 21, 2006 pa



26.2	Register Bits
26.3	Register States in Each Operating Mode.....
26.4	Register Select Conditions
Section 27 Electrical Characteristics.....	
27.1	Electrical Characteristics of H8S/2140B, H8S/2141B, H8S/2160B, and H8S/2161B
27.1.1	Absolute Maximum Ratings
27.1.2	DC Characteristics
27.1.3	AC Characteristics
27.1.4	A/D Conversion Characteristics.....
27.1.5	D/A Conversion Characteristics.....
27.1.6	Flash Memory Characteristics
27.1.7	Usage Note.....
27.2	Electrical Characteristics of H8S/2145B and H8S/2148B
27.2.1	Absolute Maximum Ratings
27.2.2	DC Characteristics
27.2.3	AC Characteristics
27.2.4	A/D Conversion Characteristics.....
27.2.5	D/A Conversion Characteristics.....
27.2.6	Flash Memory Characteristics
27.2.7	Usage Notes
27.3	Timing Chart.....
27.3.1	Clock Timing
27.3.2	Control Signal Timing
27.3.3	Bus Timing
27.3.4	On-Chip Peripheral Module Timing.....
Appendix A I/O Port States in Each Processing State	
Appendix B Product Codes	
Appendix C Package Dimensions.....	

Rev. 3.00 Mar 21, 2006 pag

RENESAS

Section 2 CPU

Figure 2.1	Exception Vector Table (Normal Mode)
Figure 2.2	Stack Structure in Normal Mode
Figure 2.3	Exception Vector Table (Advanced Mode)
Figure 2.4	Stack Structure in Advanced Mode
Figure 2.5	Memory Map
Figure 2.6	CPU Internal Registers.....
Figure 2.7	Usage of General Registers.....
Figure 2.8	Stack
Figure 2.9	General Register Data Formats (1)
Figure 2.9	General Register Data Formats (2)
Figure 2.10	Memory Data Formats
Figure 2.11	Instruction Formats (Examples).....
Figure 2.12	Branch Address Specification in Memory Indirect Addressing Mode
Figure 2.13	State Transitions.....

Section 3 MCU Operating Modes

Figure 3.1	Address Map for H8S/2140B and H8S/2160B (1).....
Figure 3.2	Address Map for H8S/2140B and H8S/2160B (2).....
Figure 3.3	Address Map for H8S/2141B and H8S/2161B (1).....
Figure 3.4	Address Map for H8S/2141B and H8S/2161B (2).....
Figure 3.5	Address Map for H8S/2145BV (1)
Figure 3.6	Address Map for H8S/2145BV (2)
Figure 3.7	Address Map for H8S/2145B (1)
Figure 3.8	Address Map for H8S/2145B (2)
Figure 3.9	Address Map for H8S/2148B (1)
Figure 3.10	Address Map for H8S/2148B (2)

Section 4 Exception Handling

Figure 4.1	Reset Sequence (Mode 3)
Figure 4.2	Stack Status after Exception Handling

	Mode 0
Figure 5.5	State Transition in Interrupt Control Mode 1
Figure 5.6	Flowchart of Procedure up to Interrupt Acceptance in Interrupt Control Mode 1
Figure 5.7	Interrupt Exception Handling.....
Figure 5.8	DTC and Interrupt Controller
Figure 5.9	Address Break Block Diagram
Figure 5.10	Address Break Timing Example
Figure 5.11	Conflict between Interrupt Generation and Disabling

Section 6 Bus Controller (BSC)

Figure 6.1	Block Diagram of Bus Controller
Figure 6.2	$\overline{\text{IOS}}$ Signal Output Timing
Figure 6.3	Access Sizes and Data Alignment Control (8-Bit Access Space).....
Figure 6.4	Access Sizes and Data Alignment Control (16-bit Access Space).....
Figure 6.5	Bus Timing for 8-Bit, 2-State Access Space.....
Figure 6.6	Bus Timing for 8-Bit, 3-State Access Space.....
Figure 6.7	Bus Timing for 16-Bit, 2-State Access Space (Even Byte Access)
Figure 6.8	Bus Timing for 16-Bit, 2-State Access Space (Odd Byte Access).....
Figure 6.9	Bus Timing for 16-Bit, 2-State Access Space (Word Access).....
Figure 6.10	Bus Timing for 16-Bit, 3-State Access Space (Even Byte Access)
Figure 6.11	Bus Timing for 16-Bit, 3-State Access Space (Odd Byte Access).....
Figure 6.12	Bus Timing for 16-Bit, 3-State Access Space (Word Access).....
Figure 6.13	Example of Wait State Insertion Timing (Pin Wait Mode).....
Figure 6.14	Access Timing Example in Burst ROM Space (AST = BRSTS1 = 1) ...
Figure 6.15	Access Timing Example in Burst ROM Space (AST = BRSTS1 = 0) ...
Figure 6.16	Examples of Idle Cycle Operation

Section 7 Data Transfer Controller (DTC)

Figure 7.1	Block Diagram of DTC.....
Figure 7.2	Block Diagram of DTC Activation Source Control.....
Figure 7.3	DTC Register Information Location in Address Space.....

Rev. 3.00 Mar 21, 2006 pa

RENESAS

Section 9 8-Bit PWM Timer (PWM)

- Figure 9.1 Block Diagram of PWM Timer
- Figure 9.2 Example of Additional Pulse Timing (when Upper 4 Bits of PWDR = 10)

Section 10 14-Bit PWM Timer (PWMX)

- Figure 10.1 PWM (D/A) Block Diagram
- Figure 10.2 PWM D/A Operation
- Figure 10.3 Output Waveform (OS = 0, DADR Corresponds to T_L)
- Figure 10.4 Output Waveform (OS = 1, DADR Corresponds to T_H)
- Figure 10.5 D/A Data Register Configuration when CFS = 1
- Figure 10.6 Output Waveform when DADR = H'0207 (OS = 1)

Section 11 16-Bit Free-Running Timer (FRT)

- Figure 11.1 Block Diagram of 16-Bit Free-Running Timer
- Figure 11.2 Example of Pulse Output
- Figure 11.3 Increment Timing with Internal Clock Source
- Figure 11.4 Increment Timing with External Clock Source
- Figure 11.5 Timing of Output Compare A Output
- Figure 11.6 Clearing of FRC by Compare-Match A Signal
- Figure 11.7 Input Capture Input Signal Timing (Usual Case)
- Figure 11.8 Input Capture Input Signal Timing (When ICRA to ICRD are Read)
- Figure 11.9 Buffered Input Capture Timing
- Figure 11.10 Buffered Input Capture Timing (BUFEA = 1)
- Figure 11.11 Timing of Input Capture Flag (ICFA, ICFB, ICFC, or ICFD) Setting
- Figure 11.12 Timing of Output Compare Flag (OCFA or OCFB) Setting
- Figure 11.13 Timing of Overflow Flag (OVF) Setting
- Figure 11.14 OCRA Automatic Addition Timing
- Figure 11.15 Timing of Input Capture Mask Signal Setting
- Figure 11.16 Timing of Input Capture Mask Signal Clearing
- Figure 11.17 FRC Write-Clear Conflict
- Figure 11.18 FRC Write-Increment Conflict

Rev. 3.00 Mar 21, 2006 page xl of liv

RENESAS

Figure 12.4	Count Timing for Internal Clock Input
Figure 12.5	Count Timing for External Clock Input (Both Edges)
Figure 12.6	Timing of CMF Setting at Compare-Match
Figure 12.7	Timing of Toggled Timer Output by Compare-Match A Signal
Figure 12.8	Timing of Counter Clear by Compare-Match
Figure 12.9	Timing of Counter Clear by External Reset Input
Figure 12.10	Timing of OVF Flag Setting
Figure 12.11	Timing of Input Capture Operation
Figure 12.12	Timing of Input Capture Signal (Input Capture Signal Is Input during T and TICRF Read)
Figure 12.13	Input Capture Signal Selection
Figure 12.14	Conflict between TCNT Write and Clear
Figure 12.15	Conflict between TCNT Write and Increment
Figure 12.16	Conflict between TCOR Write and Compare-Match

Section 13 Timer Connection

Figure 13.1	Block Diagram of Timer Connection
Figure 13.2	Timing Chart for PWM Decoding
Figure 13.3	Timing Chart for Clamp Waveform Generation (CL1 and CL2 Signals)
Figure 13.4	Timing Chart for Clamp Waveform Generation (CL3 Signal)
Figure 13.5	Timing Chart for Measurement of IVI Signal and IHI Signal Divided Waveform Periods
Figure 13.6	2fH Modification Timing Chart
Figure 13.7	Fall Modification and IHI Synchronization Timing Chart
Figure 13.8	IVG Signal/IHG Signal/CL4 Signal Timing Chart
Figure 13.9	CBLANK Output Waveform Generation

Section 14 Watchdog Timer (WDT)

Figure 14.1	Block Diagram of WDT
Figure 14.2	Watchdog Timer Mode ($\overline{\text{RST/NMI}} = 1$) Operation
Figure 14.3	Interval Timer Mode Operation
Figure 14.4	OVF Flag Set Timing

Rev. 3.00 Mar 21, 2006

RENESAS

Figure 15.3	Receive Data Sampling Timing in Asynchronous Mode
Figure 15.4	Relation between Output Clock and Transmit Data Phase (Asynchronous Mode)
Figure 15.5	Sample SCI Initialization Flowchart
Figure 15.6	Example of SCI Transmit Operation in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit).....
Figure 15.7	Sample Serial Transmission Flowchart.....
Figure 15.8	Example of SCI Receive Operation in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit).....
Figure 15.9	Sample Serial Reception Flowchart (1)
Figure 15.9	Sample Serial Reception Flowchart (2)
Figure 15.10	Example of Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A).....
Figure 15.11	Sample Multiprocessor Serial Transmission Flowchart.....
Figure 15.12	Example of SCI Receive Operation (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)
Figure 15.13	Sample Multiprocessor Serial Reception Flowchart (1)
Figure 15.13	Sample Multiprocessor Serial Reception Flowchart (2)
Figure 15.14	Data Format in Clocked Synchronous Communication (LSB-First)
Figure 15.15	Sample SCI Initialization Flowchart.....
Figure 15.16	Example of SCI Transmit Operation in Clocked Synchronous Mode
Figure 15.17	Sample Serial Transmission Flowchart.....
Figure 15.18	Example of SCI Receive Operation in Clocked Synchronous Mode.....
Figure 15.19	Sample Serial Reception Flowchart.....
Figure 15.20	Sample Flowchart of Simultaneous Serial Transmission and Reception
Figure 15.21	IrDA Block Diagram.....
Figure 15.22	IrDA Transmission and Reception.....
Figure 15.23	Example of Transmission Using DTC in Clocked Synchronous Mode.....
Figure 15.24	Sample Flowchart for Mode Transition during Transmission
Figure 15.25	Pin States during Transmission in Asynchronous Mode (Internal Clock)
Figure 15.26	Pin States during Transmission in Clocked Synchronous Mode (Internal Clock).....

Figure 16.5	I ² C Bus Data Format (Serial Format)
Figure 16.6	I ² C Bus Timing
Figure 16.7	Sample Flowchart for IIC Initialization
Figure 16.8	Sample Flowchart for Operations in Master Transmit Mode.....
Figure 16.9	Example of Operation Timing in Master Transmit Mode (MLS = WAIT
Figure 16.10	Example of Stop Condition Issuance Operation Timing in Master Tran Mode (MLS = WAIT = 0)
Figure 16.11	Sample Flowchart for Operations in Master Receive Mode (HNDS = 1
Figure 16.12	Example of Operation Timing in Master Receive Mode (MLS = WAIT = 0, HNDS = 1).....
Figure 16.13	Example of Stop Condition Issuance Operation Timing in Master Rece Mode (MLS = WAIT = 0, HNDS = 1)
Figure 16.14	Sample Flowchart for Operations in Master Receive Mode (Receiving Multiple Bytes) (WAIT = 1).....
Figure 16.15	Sample Flowchart for Operations in Master Receive Mode (Receiving a Single Byte) (WAIT = 1).....
Figure 16.16	Example of Master Receive Mode Operation Timing (MLS = ACKB = 0, WAIT = 1)
Figure 16.17	Example of Stop Condition Issuance Timing in Master Receive Mode (MLS = ACKB = 0, WAIT = 1)
Figure 16.18	Sample Flowchart for Operations in Slave Receive Mode (HNDS = 1).....
Figure 16.19	Example of Slave Receive Mode Operation Timing (1) (MLS = 0, HND
Figure 16.20	Example of Slave Receive Mode Operation Timing (2) (MLS = 0, HND
Figure 16.21	Sample Flowchart for Operations in Slave Receive Mode (HNDS = 0).....
Figure 16.22	Example of Slave Receive Mode Operation Timing (1) (MLS = ACKB = 0, HNDS = 0)
Figure 16.23	Example of Slave Receive Mode Operation Timing (2) (MLS = ACKB = 0, HNDS = 0)
Figure 16.24	Sample Flowchart for Slave Transmit Mode
Figure 16.25	Example of Slave Transmit Mode Operation Timing (MLS = 0).....
Figure 16.26	IRIC Setting Timing and SCL Control (1).....
Figure 16.27	IRIC Setting Timing and SCL Control (2).....

Rev. 3.00 Mar 21, 2006 p



Figure 16.36	TRS Bit Set Timing in Slave Mode
Figure 16.37	Diagram of Erroneous Operation when Arbitration is Lost
Section 17 Keyboard Buffer Controller	
Figure 17.1	Block Diagram of Keyboard Buffer Controller
Figure 17.2	Keyboard Buffer Controller Connection.....
Figure 17.3	Sample Receive Processing Flowchart
Figure 17.4	Receive Timing.....
Figure 17.5 (1)	Sample Transmit Processing Flowchart.....
Figure 17.5 (2)	Sample Transmit Processing Flowchart.....
Figure 17.6	Transmit Timing
Figure 17.7 (1)	Sample Receive Abort Processing Flowchart
Figure 17.7 (2)	Sample Receive Abort Processing Flowchart
Figure 17.8	Receive Abort and Transmit Start (Transmission/Reception Switchover) Timing.....
Figure 17.9	KCLKI and KDI Read Timing.....
Figure 17.10	KCLKO and KDO Write Timing.....
Figure 17.11	KBF Setting and KCLK Automatic I/O Inhibit Generation Timing.....
Figure 17.12	Receive Counter and KBBR Data Load Timing
Figure 17.13	Example of KCLK Input Fall Interrupt Operation.....
Figure 17.14	KBIOE Setting and KCLK Falling Edge Detection Timing

Section 18 Host Interface X-Bus Interface (XBS)

Figure 18.1	Block Diagram of XBS
Figure 18.2	GA20 Output.....
Figure 18.3	HIRQ Output Flowchart (Example of Channels 1 and 2).....

Section 19 Host Interface LPC Interface (LPC)

Figure 19.1	Block Diagram of LPC
Figure 19.2	Typical <u>LFRAME</u> Timing
Figure 19.3	Abort Mechanism
Figure 19.4	GA20 Output.....



Section 21 A/D Converter

Figure 21.1	Block Diagram of A/D Converter
Figure 21.2	Example of A/D Converter Operation (Scan Mode, Channels AN0 to AN7 Selected).....
Figure 21.3	A/D Conversion Timing
Figure 21.4	External Trigger Input Timing.....
Figure 21.5	A/D Conversion Accuracy Definitions
Figure 21.6	A/D Conversion Accuracy Definitions
Figure 21.7	Example of Analog Input Circuit.....
Figure 21.8	Example of Analog Input Protection Circuit
Figure 21.9	Equivalent Circuit of Analog Input Pin

Section 23 ROM

Figure 23.1	Block Diagram of Flash Memory
Figure 23.2	Flash Memory State Transitions
Figure 23.3	Boot Mode
Figure 23.4	User Program Mode (Example).....
Figure 23.5	64-Kbyte Flash Memory Block Configuration
Figure 23.6	128-Kbyte Flash Memory Block Configuration
Figure 23.7	256-Kbyte Flash Memory Block Configuration
Figure 23.8	On-Chip RAM Area in Boot Mode.....
Figure 23.9	ID Code Area
Figure 23.10	Programming/Erasing Flowchart Example in User Program Mode.....
Figure 23.11	Program/Program-Verify Flowchart
Figure 23.12	Erase/Erase-Verify Flowchart.....
Figure 23.13	Memory Map in Programmer Mode

Section 24 Clock Pulse Generator

Figure 24.1	Block Diagram of Clock Pulse Generator.....
Figure 24.2	Typical Connection to Crystal Resonator
Figure 24.3	Equivalent Circuit of Crystal Resonator
Figure 24.4	Example of External Clock Input.....

Rev. 3.00 Mar 21, 2006 p

RENESAS

Figure 25.3	Application Example in Software Standby Mode.....
Figure 25.4	Hardware Standby Mode Timing.....

Section 27 Electrical Characteristics

Figure 27.1	Darlington Pair Drive Circuit (Example).....
Figure 27.2	LED Drive Circuit (Example).....
Figure 27.3	Output Load Circuit
Figure 27.4	Connection of VCL Capacitor
Figure 27.5	Connection of VCL Capacitor
Figure 27.6	System Clock Timing
Figure 27.7	Oscillation Settling Timing.....
Figure 27.8	Oscillation Settling Timing (Exiting Software Standby Mode).....
Figure 27.9	Reset Input Timing
Figure 27.10	Interrupt Input Timing
Figure 27.11	Basic Bus Timing (Two-State Access)
Figure 27.12	Basic Bus Timing (Three-State Access)
Figure 27.13	Basic Bus Timing (Three-State Access with One Wait State).....
Figure 27.14	Burst ROM Access Timing (Two-State Access)
Figure 27.15	Burst ROM Access Timing (One-State Access).....
Figure 27.16	I/O Port Input/Output Timing
Figure 27.17	FRT Input/Output Timing.....
Figure 27.18	FRT Clock Input Timing
Figure 27.19	8-Bit Timer Output Timing.....
Figure 27.20	8-Bit Timer Clock Input Timing.....
Figure 27.21	8-Bit Timer Reset Input Timing.....
Figure 27.22	PWM, PWMX Output Timing.....
Figure 27.23	SCK Clock Input Timing
Figure 27.24	SCI Input/Output Timing (Synchronous Mode)
Figure 27.25	A/D Converter External Trigger Input Timing
Figure 27.26	WDT Output Timing ($\overline{\text{RESO}}$).....
Figure 27.27	Host Interface (XBS) Timing.....
Figure 27.28	Keyboard Buffer Controller Timing

Rev. 3.00 Mar 21, 2006 p

RENESAS

Section 2 CPU

Table 2.1	Instruction Classification.....
Table 2.2	Operation Notation.....
Table 2.3	Data Transfer Instructions.....
Table 2.4	Arithmetic Operations Instructions (1).....
Table 2.4	Arithmetic Operations Instructions (2).....
Table 2.5	Logic Operations Instructions.....
Table 2.6	Shift Instructions.....
Table 2.7	Bit Manipulation Instructions (1).....
Table 2.7	Bit Manipulation Instructions (2).....
Table 2.8	Branch Instructions.....
Table 2.9	System Control Instructions.....
Table 2.10	Block Data Transfer Instructions.....
Table 2.11	Addressing Modes.....
Table 2.12	Absolute Address Access Ranges.....
Table 2.13	Effective Address Calculation (1).....
Table 2.13	Effective Address Calculation (2).....

Section 3 MCU Operating Modes

Table 3.1	MCU Operating Mode Selection.....
Table 3.2	Pin Functions in Each Mode.....

Section 4 Exception Handling

Table 4.1	Exception Types and Priority.....
Table 4.2	Exception Handling Vector Table.....
Table 4.3	Status of CCR after Trap Instruction Exception Handling.....

Section 5 Interrupt Controller

Table 5.1	Pin Configuration.....
Table 5.2	Correspondence between Interrupt Source and ICR.....
Table 5.3	Interrupt Sources, Vector Addresses, and Interrupt Priorities.....
Table 5.4	Interrupt Control Modes.....

Rev. 3.00 Mar 21, 2006 page xviii of liv

RENESAS

Section 7 Data Transfer Controller (DTC)

Table 7.1	Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs
Table 7.2	Register Functions in Normal Mode
Table 7.3	Register Functions in Repeat Mode
Table 7.4	Register Functions in Block Transfer Mode.....
Table 7.5	DTC Execution Status
Table 7.6	Number of States Required for Each Execution Status

Section 8 I/O Ports

Table 8.1	Port Functions of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B
Table 8.2	Input Pull-Up MOS States (Port 1)
Table 8.3	Input Pull-Up MOS States (Port 2)
Table 8.4	Input Pull-Up MOS States (Port 3)
Table 8.5	Input Pull-Up MOS States (Port 6)
Table 8.6	Input Pull-Up MOS States (Port A).....
Table 8.7	Input Pull-Up MOS States (Port B).....
Table 8.8	H8S/2160B, H8S/2161B Additional Port Functions
Table 8.9	Input Pull-Up MOS States (Port C and port D).....
Table 8.10	Input Pull-Up MOS States (Port E and port F).....

Section 9 8-Bit PWM Timer (PWM)

Table 9.1	Pin Configuration
Table 9.2	Internal Clock Selection
Table 9.3	Resolution, PWM Conversion Period and Carrier Frequency when $\phi = 1$
Table 9.4	Duty Cycle of Basic Pulse
Table 9.5	Position of Pulses Added to Basic Pulses.....

Section 10 14-Bit PWM Timer (PWMX)

Table 10.1	Pin Configuration
Table 10.2	Read and Write Access Methods for 16-Bit Registers
Table 10.3	Settings and Operation (Examples when $\phi = 10$ MHz).....

Rev. 3.00 Mar 21, 2006 p

RENESAS

Table 12.2	Clock Input to TCNT and Count Condition
Table 12.3	Input Capture Signal Selection
Table 12.4	Interrupt Sources of 8-Bit Timers TMR_0, TMR_1, TMR_Y, and TMR_Z
Table 12.5	Timer Output Priorities
Table 12.6	Switching of Internal Clocks and TCNT Operation

Section 13 Timer Connection

Table 13.1	Pin Configuration
Table 13.2	Synchronization Signal Connection Enable
Table 13.3	Registers Accessible by TMR_X/TMR_Y
Table 13.4	Examples of TCR Settings
Table 13.5	Examples of TCORB (Pulse Width Threshold) Settings
Table 13.6	Examples of TCR and TCSR Settings
Table 13.7	Examples of TCR, TCSR, TOCR, and OCRDM Settings
Table 13.8	Examples of TCR, TCSR, and TCORB Settings
Table 13.9	Examples of OCRAR, OCRAF, TCORA, TCORB, TCR, and TCSR Settings
Table 13.10	HSYNCO Output Modes
Table 13.11	VSYNCO Output Modes

Section 14 Watchdog Timer (WDT)

Table 14.1	Pin Configuration
Table 14.2	WDT Interrupt Source

Section 15 Serial Communication Interface (SCI and IrDA)

Table 15.1	Pin Configuration
Table 15.2	Relationships between N Setting in BRR and Bit Rate B
Table 15.3	BRR Settings for Various Bit Rates (Asynchronous Mode)
Table 15.4	Maximum Bit Rate for Each Frequency (Asynchronous Mode)
Table 15.5	Maximum Bit Rate with External Clock Input (Asynchronous Mode)
Table 15.6	BRR Settings for Various Bit Rates (Clocked Synchronous Mode)
Table 15.7	Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode)
Table 15.8	Serial Transfer Formats (Asynchronous Mode)

Table 16.5	Flags and Transfer States (Slave Mode).....
Table 16.6	I ² C Bus Data Format Symbols.....
Table 16.7	Examples of Operation Using DTC
Table 16.8	IIC Interrupt Sources.....
Table 16.9	I ² C Bus Timing (SCL and SDA Outputs).....
Table 16.10	Permissible SCL Rise Time (t_{sr}) Values
Table 16.11	I ² C Bus Timing (with Maximum Influence of t_{sr}/t_{sp}).....

Section 17 Keyboard Buffer Controller

Table 17.1	Pin Configuration
------------	-------------------------

Section 18 Host Interface X-Bus Interface (XBS)

Table 18.1	Pin Configuration
Table 18.2	Set/Clear Timing for STR Flags.....
Table 18.3	Host Interface Channel Selection and Pin Operation
Table 18.4	Host Interface Operations from HIF Host, and Slave Operation.....
Table 18.5	GA20 (P81) Set/Clear Timing.....
Table 18.6	Fast A20 Gate Output Signal.....
Table 18.7	Scope of HIF Pin Shutdown.....
Table 18.8	Input Buffer Full Interrupts
Table 18.9	HIRQ Setting/Clearing Conditions.....

Section 19 Host Interface LPC Interface (LPC)

Table 19.1	Pin Configuration
Table 19.2	Register Selection.....
Table 19.3	GA20 (P81) Set/Clear Timing.....
Table 19.4	Fast A20 Gate Output Signals
Table 19.5	Scope of Host Interface Pin Shutdown.....
Table 19.6	Scope of Initialization in Each Host Interface Mode.....
Table 19.7	Receive Complete Interrupts and Error Interrupt
Table 19.8	HIRQ Setting and Clearing Conditions.....

Rev. 3.00 Mar 21, 2006

RENESAS

Section 23 ROM

Table 23.1	Differences between Boot Mode and User Program Mode.....
Table 23.2	Pin Configuration
Table 23.3	Operating Modes and ROM
Table 23.4	On-Board Programming Mode Settings
Table 23.5	Boot Mode Operation.....
Table 23.6	System Clock Frequencies for which Automatic Adjustment of LSI Bit R Is Possible.....

Section 24 Clock Pulse Generator

Table 24.1	Damping Resistance Values.....
Table 24.2	Crystal Resonator Parameters
Table 24.3	External Clock Input Conditions.....
Table 24.4	External Clock Output Stabilization Delay Time.....
Table 24.5	Subclock Input Conditions

Section 25 Power-Down Modes

Table 25.1	Operating Frequency and Wait Time
Table 25.2	LSI Internal States in Each Mode.....

Section 27 Electrical Characteristics

Table 27.1	Absolute Maximum Ratings.....
Table 27.2	DC Characteristics (1).....
Table 27.2	DC Characteristics (2).....
Table 27.2	DC Characteristics (3) When LPC Function Is Used
Table 27.3	Permissible Output Currents
Table 27.4	Bus Drive Characteristics.....
Table 27.5	Clock Timing
Table 27.6	Control Signal Timing.....
Table 27.7	Bus Timing (1) (Normal Mode)
Table 27.7	Bus Timing (2) (Advanced Mode)
Table 27.8	Timing of On-Chip Peripheral Modules (1).....

Table 27.15	Flash Memory Characteristics.....
Table 27.16	Absolute Maximum Ratings.....
Table 27.17	DC Characteristics (1).....
Table 27.17	DC Characteristics (2).....
Table 27.17	DC Characteristics (3).....
Table 27.17	DC Characteristics (4).....
Table 27.17	DC Characteristics (5).....
Table 27.17	DC Characteristics (6).....
Table 27.17	DC Characteristics (7) (3-V Version of H8S/2145BV) When LPC Function Is Used.....
Table 27.18	Permissible Output Currents
Table 27.19	Bus Drive Characteristics
Table 27.20	Clock Timing
Table 27.21	Control Signal Timing.....
Table 27.22	Bus Timing (1) (Normal Mode)
Table 27.22	Bus Timing (2) (Advanced Mode)
Table 27.23	Timing of On-Chip Peripheral Modules (1).....
Table 27.23	Timing of On-Chip Peripheral Modules (2).....
Table 27.24	Keyboard Buffer Controller Timing.....
Table 27.25	I ² C Bus Timing.....
Table 27.26	LPC Module Timing (For H8S/2145B Only).....
Table 27.27	A/D Conversion Characteristics (AN7 to AN0 Input: 134/266-State Conversion)
Table 27.28	A/D Conversion Characteristics (CIN15 to CIN0 Input: 134/266-State Conversion)
Table 27.29	D/A Conversion Characteristics
Table 27.30	Flash Memory Characteristics (Operation Range at Programming/Erasing).....

Appendix A I/O Port States in Each Processing State

Table A.1	I/O Port States in Each Processing State
-----------	--

Rev. 3.00 Mar 21, 2006

RENESAS

- 65 basic instructions
- Various peripheral functions
 - Data transfer controller (DTC)
 - 8-bit PWM timer (PWM)
 - 14-bit PWM timer (PWMX)
 - 16-bit free-running timer (FRT)
 - 8-bit timer (TMR)
 - Timer connection
 - Watchdog timer (WDT)
 - Asynchronous or clocked synchronous serial communication interface (SCI, IrDA)
 - I²C bus interface (IIC)
 - Keyboard buffer controller
 - Host interface X-BUS interface (XBS)
 - Host interface LPC interface (LPC)*
 - 8-bit D/A converter
 - 10-bit A/D converter
 - Clock pulse generator

Note: * The LPC function is not supported by H8S/2148B and H8S/2145B (5-V ve

HD64F2145B	256 kbytes	8 kbytes
HD64F2148BV*	128 kbytes	4 kbytes
HD64F2148B	128 kbytes	4 kbytes

Note: * 3-V version product

- General I/O ports
I/O pins: 74 (H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B)
I/O pins: 114 (H8S/2160B and H8S/2161B)
Input-only pins: 8
- Supports various power-down states
- Compact package

Product	Package	Code	Body Size	Pin
H8S/2161B, H8S/2160B	TQFP-144	TFP-144	16.0 × 16.0 mm	0.4
H8S/2141B, H8S/2140B	QFP-100B	FP-100B	14.0 × 14.0 mm	0.5
H8S/2145B, H8S/2148B	TQFP-100B	TFP-100B		

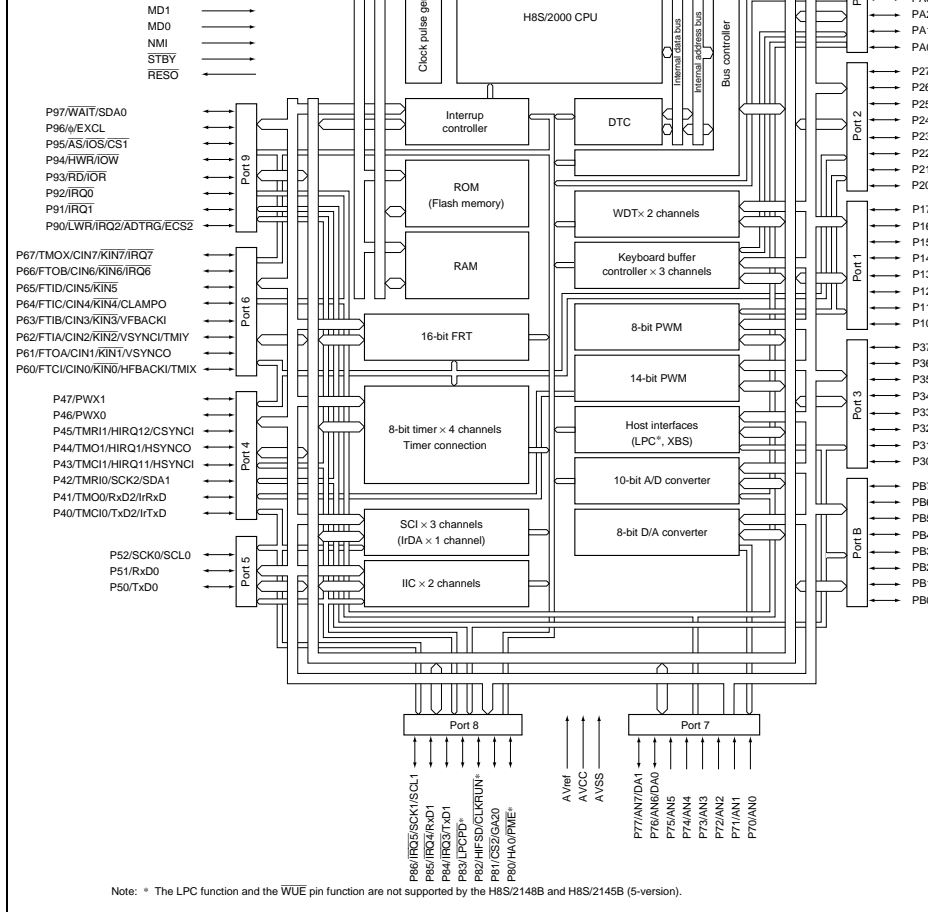


Figure 1.1 Internal Block Diagram of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2148B

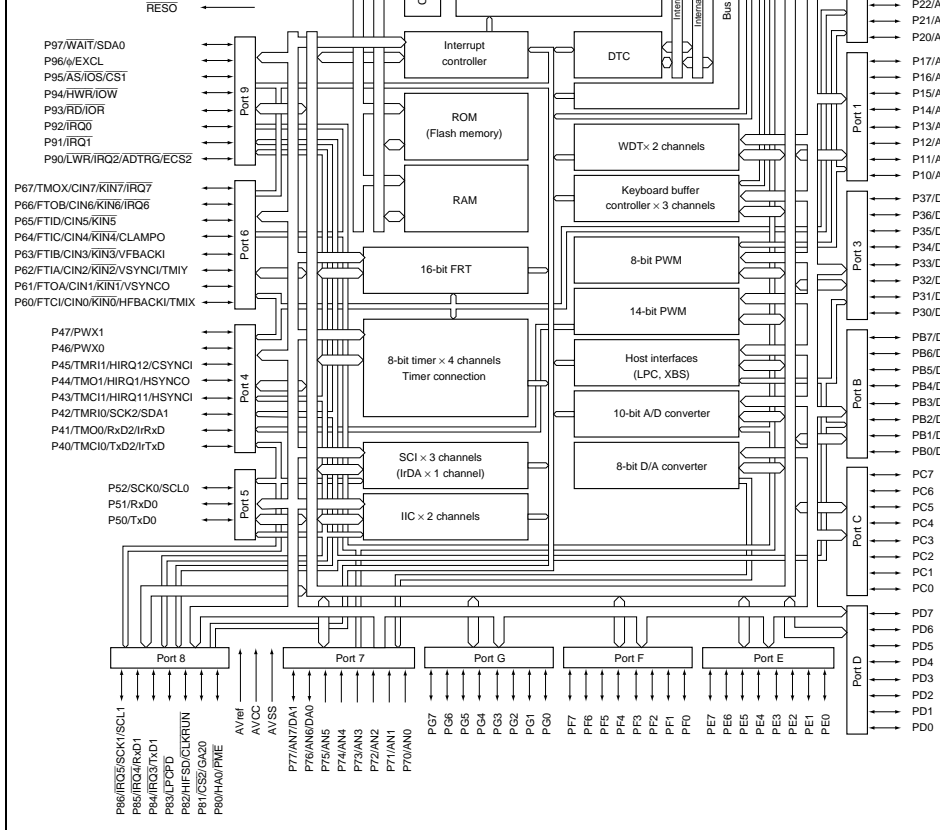


Figure 1.2 Internal Block Diagram of H8S/2160B and H8S/2161B



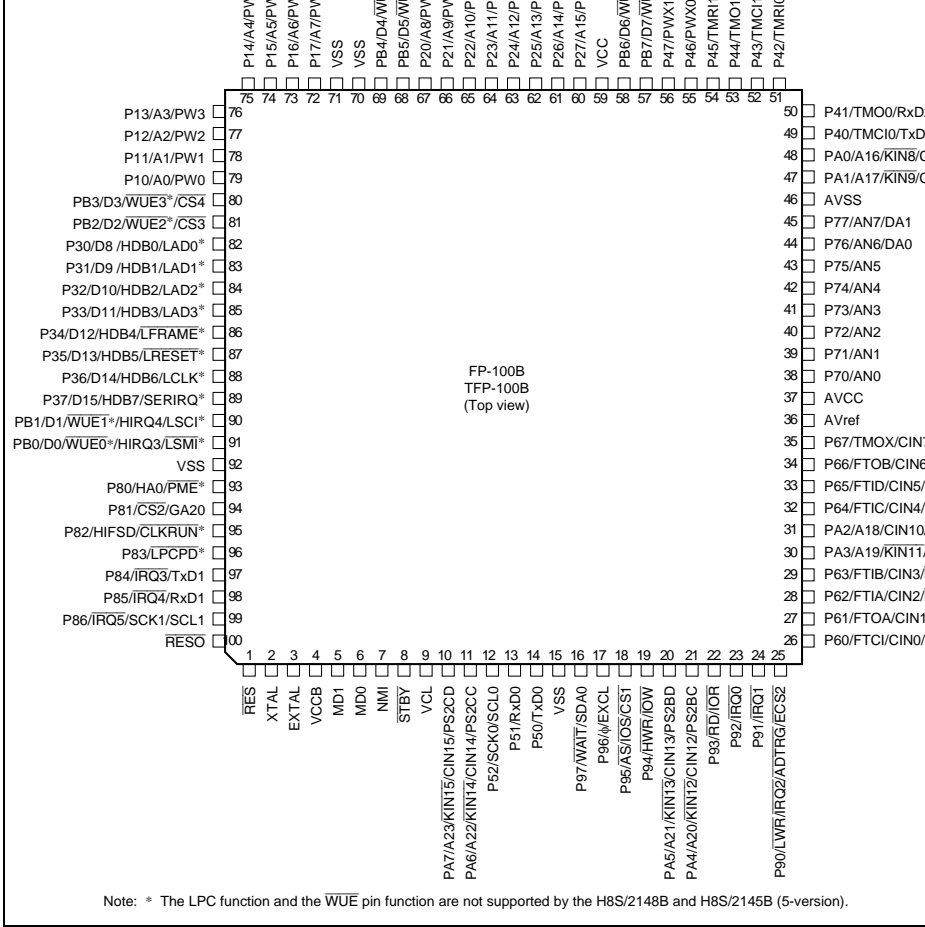


Figure 1.3 Pin Arrangement of H8S/2140B, H8S/2141B, H8S/2145B, and H8S/2145B (5-version)

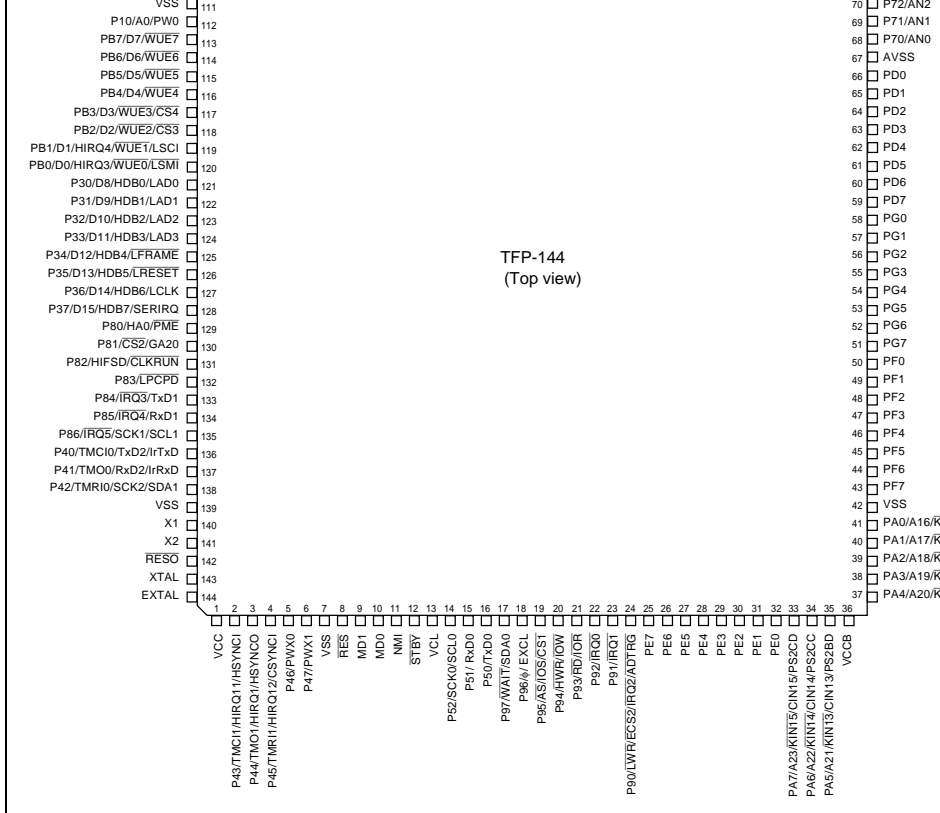


Figure 1.4 Pin Arrangement of H8S/2160B and H8S/2161B



1	RES	RES	RES	RES
2	XTAL	XTAL	XTAL	XTAL
3	EXTAL	EXTAL	EXTAL	EXTAL
4	VCCB	VCCB	VCCB	VCCB
5	MD1	MD1	MD1	MD1
6	MD0	MD0	MD0	MD0
7	NMI	NMI	NMI	NMI
8	STBY	STBY	STBY	STBY
9	VCL	VCL	VCL	VCL
10 (B)	PA7/CIN15/ $\overline{\text{KIN15}}$ / PS2CD	PA7/A23/CIN15/ $\overline{\text{KIN15}}$ /PS2CD	PA7/CIN15/ $\overline{\text{KIN15}}$ / PS2CD	PA7/CIN15/ $\overline{\text{KIN15}}$ / PS2CD
11 (B)	PA6/CIN14/ $\overline{\text{KIN14}}$ / PS2CC	PA6/A22/CIN14/ $\overline{\text{KIN14}}$ /PS2CC	PA6/CIN14/ $\overline{\text{KIN14}}$ / PS2CC	PA6/CIN14/ $\overline{\text{KIN14}}$ / PS2CC
12 (N)	P52/SCK0/SCL0	P52/SCK0/SCL0	P52/SCK0/SCL0	P52/SCK0/SCL0
13	P51/RxD0	P51/RxD0	P51/RxD0	P51/RxD0
14	P50/TxD0	P50/TxD0	P50/TxD0	P50/TxD0
15	VSS	VSS	VSS	VSS
16 (N)	P97/ $\overline{\text{WAIT}}$ /SDA0	P97/ $\overline{\text{WAIT}}$ /SDA0	P97/SDA0	P97/SDA0
17	P96/ ϕ /EXCL	P96/ ϕ /EXCL	P96/ ϕ /EXCL	P96/ ϕ /EXCL
18	$\overline{\text{AS}}$ / $\overline{\text{IOS}}$	$\overline{\text{AS}}$ / $\overline{\text{IOS}}$	P95/CS1	P95/CS1
19	$\overline{\text{HWR}}$	$\overline{\text{HWR}}$	P94/ $\overline{\text{IOW}}$	P94/ $\overline{\text{IOW}}$
20 (B)	PA5/CIN13/ $\overline{\text{KIN13}}$ / PS2BD	PA5/A21/CIN13/ $\overline{\text{KIN13}}$ /PS2BD	PA5/CIN13/ $\overline{\text{KIN13}}$ / PS2BD	PA5/CIN13/ $\overline{\text{KIN13}}$ / PS2BD
21 (B)	PA4/CIN12/ $\overline{\text{KIN12}}$ / PS2BC	PA4/A20/CIN12/ $\overline{\text{KIN12}}$ /PS2BC	PA4/CIN12/ $\overline{\text{KIN12}}$ / PS2BC	PA4/CIN12/ $\overline{\text{KIN12}}$ / PS2BC

25	P90/LWR/IRQ2/ ADTRG	P90/LWR/IRQ2/ ADTRG	P90/ECS2/IRQ2/ ADTRG	VC
26	P60/FTCI/CIN0/ KIN0/HFBACKI/ TMIX	P60/FTCI/CIN0/ KIN0/HFBACKI/ TMIX	P60/FTCI/CIN0/ KIN0/HFBACKI/ TMIX	NC
27	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	NC
28	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	NC
29	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	NC
30 (B)	PA3/CIN11/KIN11/ PS2AD	PA3/A19/CIN11/ KIN11/PS2AD	PA3/CIN11/KIN11/ PS2AD	NC
31 (B)	PA2/CIN10/KIN10/ PS2AC	PA2/A18/CIN10/ KIN10/PS2AC	PA2/CIN10/KIN10/ PS2AC	NC
32	P64/FTIC/CIN4/ KIN4/CLAMPO	P64/FTIC/CIN4/ KIN4/CLAMPO	P64/FTIC/CIN4/ KIN4/CLAMPO	NC
33	P65/FTID/CIN5/ KIN5	P65/FTID/CIN5/ KIN5	P65/FTID/CIN5/ KIN5	NC
34	P66/FTOB/CIN6/ KIN6/IRQ6	P66/FTOB/CIN6/ KIN6/IRQ6	P66/FTOB/CIN6/ KIN6/IRQ6	NC
35	P67/TMOX/CIN7/ KIN7/IRQ7	P67/TMOX/CIN7/ KIN7/IRQ7	P67/TMOX/CIN7/ KIN7/IRQ7	VS
36	AVref	AVref	AVref	VC
37	AVCC	AVCC	AVCC	VC
38	P70/AN0	P70/AN0	P70/AN0	NC
39	P71/AN1	P71/AN1	P71/AN1	NC
40	P72/AN2	P72/AN2	P72/AN2	NC
41	P73/AN3	P73/AN3	P73/AN3	NC

45	P77/AN7/DA1	P77/AN7/DA1	P77/AN7/DA1	NC
46	AVSS	AVSS	AVSS	VS
47 (B)	PA1/CIN9/KIN9	PA1/A17/CIN9/KIN9	PA1/CIN9/KIN9	NC
48 (B)	PA0/CIN8/KIN8	PA0/A16/CIN8/KIN8	PA0/CIN8/KIN8	NC
49	P40/TMC10/TxD2/ IrTxD	P40/TMC10/TxD2/ IrTxD	P40/TMC10/TxD2/ IrTxD	NC
50	P41/TMO0/RxD2/ IrRxD	P41/TMO0/RxD2/ IrRxD	P41/TMO0/RxD2/ IrRxD	NC
51 (N)	P42/TMRI0/SCK2/ SDA1	P42/TMRI0/SCK2/ SDA1	P42/TMRI0/SCK2/ SDA1	NC
52	P43/TMC11/ HSYNCl	P43/TMC11/ HSYNCl	P43/TMC11/HIRQ11/ HSYNCl	NC
53	P44/TMO1/ HSYNCO	P44/TMO1/ HSYNCO	P44/TMO1/HIRQ1/ HSYNCO	NC
54	P45/TMRI1/ CSYNCl	P45/TMRI1/ CSYNCl	P45/TMRI1/HIRQ12/ CSYNCl	NC
55	P46/PWX0	P46/PWX0	P46/PWX0	NC
56	P47/PWX1	P47/PWX1	P47/PWX1	NC
57	PB7/D7/WUE7*	PB7/D7/WUE7*	PB7/WUE7*	NC
58	PB6/D6/WUE6*	PB6/D6/WUE6*	PB6/WUE6*	NC
59	VCC	VCC	VCC	VC
60	A15	P27/A15/PW15/ CBLANK	P27/PW15/ CBLANK	CE
61	A14	P26/A14/PW14	P26/PW14	FA
62	A13	P25/A13/PW13	P25/PW13	FA
63	A12	P24/A12/PW12	P24/PW12	FA
64	A11	P23/A11/PW11	P23/PW11	FA

Rev. 3.00 Mar 21, 2006 p
REJ05

RENESAS

68	PB5/D5/WUE5	PB5/D5/WUE5	PB5/WUE5	NC
69	PB4/D4/WUE4*	PB4/D4/WUE4*	PB4/WUE4*	NC
70	VSS	VSS	VSS	VSS
71	VSS	VSS	VSS	VSS
72	A7	P17/A7/PW7	P17/PW7	FA7
73	A6	P16/A6/PW6	P16/PW6	FA6
74	A5	P15/A5/PW5	P15/PW5	FA5
75	A4	P14/A4/PW4	P14/PW4	FA4
76	A3	P13/A3/PW3	P13/PW3	FA3
77	A2	P12/A2/PW2	P12/PW2	FA2
78	A1	P11/A1/PW1	P11/PW1	FA1
79	A0	P10/A0/PW0	P10/PW0	FA0
80	PB3/D3/WUE3*	PB3/D3/WUE3*	PB3/WUE3*/CS4	NC
81	PB2/D2/WUE2*	PB2/D2/WUE2*	PB2/WUE2*/CS3	NC
82	D8	D8	P30/HDB0/LAD0*	FO
83	D9	D9	P31/HDB1/LAD1*	FO
84	D10	D10	P32/HDB2/LAD2*	FO
85	D11	D11	P33/HDB3/LAD3*	FO
86	D12	D12	P34/HDB4/ LFRAME*	FO
87	D13	D13	P35/HDB5/ LRESET*	FO
88	D14	D14	P36/HDB6/LCLK*	FO
89	D15	D15	P37/HDB7/SERIRQ*	FO
90	PB1/D1/WUE1*	PB1/D1/WUE1*	PB1/HIRQ4/WUE1*/ LSCI*	NC

94	P81	P81	P81/ $\overline{CS2}$ /GA20	NC
95	P82	P82	P82/HIFSD/ \overline{CLKRUN} *	NC
96	P83	P83	P83/ \overline{LPCPD} *	NC
97	P84/ $\overline{IRQ3}$ /TxD1	P84/ $\overline{IRQ3}$ /TxD1	P84/ $\overline{IRQ3}$ /TxD1	NC
98	P85/ $\overline{IRQ4}$ /RxD1	P85/ $\overline{IRQ4}$ /RxD1	P85/ $\overline{IRQ4}$ /RxD1	NC
99 (N)	P86/ $\overline{IRQ5}$ /SCK1/ SCL1	P86/ $\overline{IRQ5}$ /SCK1/ SCL1	P86/ $\overline{IRQ5}$ /SCK1/ SCL1	NC
100	$\overline{RES0}$	$\overline{RES0}$	$\overline{RES0}$	NC

Notes: The (B) in Pin No. means the VCCB drive and the (N) in Pin No. means the NC (no connection) or pull/open-drain drive.

* The LPC function and the \overline{WUE} pin function are not supported by the H8S/2145B (5-version).

	HSYNC1	HSYNC1	HSYNC1	
3	P44/TMO1/ HSYNCO	P44/TMO1/ HSYNCO	P44/TMO1/HIRQ1/ HSYNCO	NC
4	P45/TMRI1/ CSYNCI	P45/TMRI1/ CSYNCI	P45/TMRI1/HIRQ12/ CSYNCI	NC
5	P46/PWX0	P46/PWX0	P46/PWX0	NC
6	P47/PWX1	P47/PWX1	P47/PWX1	NC
7	VSS	VSS	VSS	VSS
8	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
9	MD1	MD1	MD1	VSS
10	MD0	MD0	MD0	VSS
11	NMI	NMI	NMI	FA9
12	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	VCC
13	VCL	VCL	VCL	VCC
14 (N)	P52/SCK0/SCL0	P52/SCK0/SCL0	P52/SCK0/SCL0	FA1
15	P51/RxD0	P51/RxD0	P51/RxD0	FA1
16	P50/TxD0	P50/TxD0	P50/TxD0	NC
17 (N)	P97/ $\overline{\text{WAIT}}$ /SDA0	P97/ $\overline{\text{WAIT}}$ /SDA0	P97/SDA0	VCC
18	P96/ ϕ /EXCL	P96/ ϕ /EXCL	P96/ ϕ /EXCL	NC
19	$\overline{\text{AS}}/\overline{\text{IOS}}$	$\overline{\text{AS}}/\overline{\text{IOS}}$	P95/ $\overline{\text{CS1}}$	FA1
20	$\overline{\text{HWR}}$	$\overline{\text{HWR}}$	P94/ $\overline{\text{IOW}}$	FA1
21	$\overline{\text{RD}}$	$\overline{\text{RD}}$	P93/ $\overline{\text{IOR}}$	$\overline{\text{WE}}$
22	P92/ $\overline{\text{IRQ0}}$	P92/ $\overline{\text{IRQ0}}$	P92/ $\overline{\text{IRQ0}}$	VSS
23	P91/ $\overline{\text{IRQ1}}$	P91/ $\overline{\text{IRQ1}}$	P91/ $\overline{\text{IRQ1}}$	VCC
24	P90/ $\overline{\text{LWR}}/\overline{\text{IRQ2}}/ADTRG$	P90/ $\overline{\text{LWR}}/\overline{\text{IRQ2}}/ADTRG$	P90/ $\overline{\text{IRQ2}}/\overline{\text{ADTRG}}/ECS2$	VCC

28	PE4	PE4	PE4	NC
29	PE3	PE3	PE3	NC
30	PE2	PE2	PE2	NC
31	PE1	PE1	PE1	NC
32	PE0	PE0	PE0	NC
33 (B)	PA7/CIN15/ $\overline{\text{KIN15}}$ / PS2CD	PA7/A23/CIN15/ $\overline{\text{KIN15}}$ /PS2CD	PA7/CIN15/ $\overline{\text{KIN15}}$ / PS2CD	NC
34 (B)	PA6/CIN14/ $\overline{\text{KIN14}}$ / PS2CC	PA6/A22/CIN14/ $\overline{\text{KIN14}}$ /PS2CC	PA6/CIN14/ $\overline{\text{KIN14}}$ / PS2CC	NC
35 (B)	PA5/CIN13/ $\overline{\text{KIN13}}$ / PS2BD	PA5/A21/CIN13/ $\overline{\text{KIN13}}$ /PS2BD	PA5/CIN13/ $\overline{\text{KIN13}}$ / PS2BD	NC
36	VCCB	VCCB	VCCB	VO
37 (B)	PA4/CIN12/ $\overline{\text{KIN12}}$ / PS2BC	PA4/A20/CIN12/ $\overline{\text{KIN12}}$ /PS2BC	PA4/CIN12/ $\overline{\text{KIN12}}$ / PS2BC	NC
38 (B)	PA3/CIN11/ $\overline{\text{KIN11}}$ / PS2AD	PA3/A19/CIN11/ $\overline{\text{KIN11}}$ /PS2AD	PA3/CIN11/ $\overline{\text{KIN11}}$ / PS2AD	NC
39 (B)	PA2/CIN10/ $\overline{\text{KIN10}}$ / PS2AC	PA2/A18/CIN10/ $\overline{\text{KIN10}}$ /PS2AC	PA2/CIN10/ $\overline{\text{KIN10}}$ / PS2AC	NC
40 (B)	PA1/CIN9/ $\overline{\text{KIN9}}$	PA1/A17/CIN9/ $\overline{\text{KIN9}}$	PA1/CIN9/ $\overline{\text{KIN9}}$	NC
41 (B)	PA0/CIN8/ $\overline{\text{KIN8}}$	PA0/A16/CIN8/ $\overline{\text{KIN8}}$	PA0/CIN8/ $\overline{\text{KIN8}}$	NC
42	VSS	VSS	VSS	VS
43	PF7	PF7	PF7	NC
44	PF6	PF6	PF6	NC
45	PF5	PF5	PF5	NC
46	PF4	PF4	PF4	NC
47	PF3	PF3	PF3	NC
48	PF2	PF2	PF2	NC

52 (N)	PG6	PG6	PG6	NC
53 (N)	PG5	PG5	PG5	NC
54 (N)	PG4	PG4	PG4	NC
55 (N)	PG3	PG3	PG3	NC
56 (N)	PG2	PG2	PG2	NC
57 (N)	PG1	PG1	PG1	NC
58 (N)	PG0	PG0	PG0	NC
59	PD7	PD7	PD7	NC
60	PD6	PD6	PD6	NC
61	PD5	PD5	PD5	NC
62	PD4	PD4	PD4	NC
63	PD3	PD3	PD3	NC
64	PD2	PD2	PD2	NC
65	PD1	PD1	PD1	NC
66	PD0	PD0	PD0	NC
67	AVSS	AVSS	AVSS	VSS
68	P70/AN0	P70/AN0	P70/AN0	NC
69	P71/AN1	P71/AN1	P71/AN1	NC
70	P72/AN2	P72/AN2	P72/AN2	NC
71	P73/AN3	P73/AN3	P73/AN3	NC
72	P74/AN4	P74/AN4	P74/AN4	NC
73	P75/AN5	P75/AN5	P75/AN5	NC
74	P76/AN6/DA0	P76/AN6/DA0	P76/AN6/DA0	NC
75	P77/AN7/DA1	P77/AN7/DA1	P77/AN7/DA1	NC
76	AVCC	AVCC	AVCC	VCC

79	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	NC
80	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	P62/FTIA/CIN2/ KIN2/VSYNCI/TMIY	NC
81	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	NC
82	P64/FTIC/CIN4/ KIN4/CLAMPO	P64/FTIC/CIN4/ KIN4/CLAMPO	P64/FTIC/CIN4/ KIN4/CLAMPO	NC
83	P65/FTID/CIN5/ KIN5	P65/FTID/CIN5/ KIN5	P65/FTID/CIN5/ KIN5	NC
84	P66/FTOB/CIN6/ KIN6/IRQ6	P66/FTOB/CIN6/ KIN6/IRQ6	P66/FTOB/CIN6/ KIN6/IRQ6	NC
85	P67/TMOX/CIN7/ KIN7/IRQ7	P67/TMOX/CIN7/ KIN7/IRQ7	P67/TMOX/CIN7/ KIN7/IRQ7	VS
86	VCC	VCC	VCC	VC
87	PC7	PC7	PC7	NC
88	PC6	PC6	PC6	NC
89	PC5	PC5	PC5	NC
90	PC4	PC4	PC4	NC
91	PC3	PC3	PC3	NC
92	PC2	PC2	PC2	NC
93	PC1	PC1	PC1	NC
94	PC0	PC0	PC0	NC
95	VSS	VSS	VSS	VS
96	A15	P27/A15/PW15/ CBLANK	P27/PW15/ CBLANK	CB
97	A14	P26/A14/PW14	P26/PW14	FA

101	A10	P22/A10/PW10	P22/PW10	FA10
102	A9	P21/A9/PW9	P21/PW9	FA9
103	A8	P20/A8/PW8	P20/PW8	FA8
104	A7	P17/A7/PW7	P17/PW7	FA7
105	A6	P16/A6/PW6	P16/PW6	FA6
106	A5	P15/A5/PW5	P15/PW5	FA5
107	A4	P14/A4/PW4	P14/PW4	FA4
108	A3	P13/A3/PW3	P13/PW3	FA3
109	A2	P12/A2/PW2	P12/PW2	FA2
110	A1	P11/A1/PW1	P11/PW1	FA1
111	VSS	VSS	VSS	VSS
112	A0	P10/A0/PW0	P10/PW0	FA0
113	PB7/D7/WUE7	PB7/D7/WUE7	PB7/WUE7	NC
114	PB6/D6/WUE6	PB6/D6/WUE6	PB6/WUE6	NC
115	PB5/D5/WUE5	PB5/D5/WUE5	PB5/WUE5	NC
116	PB4/D4/WUE4	PB4/D4/WUE4	PB4/WUE4	NC
117	PB3/D3/WUE3	PB3/D3/WUE3	PB3/WUE3/CS4	NC
118	PB2/D2/WUE2	PB2/D2/WUE2	PB2/WUE2/CS3	NC
119	PB1/D1/WUE1	PB1/D1/WUE1	PB1/HIRQ4/WUE1/ LSCI	NC
120	PB0/D0/WUE0	PB0/D0/WUE0	PB0/HIRQ3/WUE0/ LSMI	NC
121	D8	D8	P30/HDB0/LAD0	FO0
122	D9	D9	P31/HDB1/LAD1	FO1
123	D10	D10	P32/HDB2/LAD2	FO2
124	D11	D11	P33/HDB3/LAD3	FO3

128	D15	D15	P37/HDB7/SERIRQ	FC
129	P80	P80	P80/HA0/PME	NC
130	P81	P81	P81/CS2/GA20	NC
131	P82	P82	P82/HIFSD/ CLKRUN	NC
132	P83	P83	P83/LPCPD	NC
133	P84/IRQ3/TxD1	P84/IRQ3/TxD1	P84/IRQ3/TxD1	NC
134	P85/IRQ4/RxD1	P85/IRQ4/RxD1	P85/IRQ4/RxD1	NC
135 (N)	P86/IRQ5/SCK1/ SCL1	P86/IRQ5/SCK1/ SCL1	P86/IRQ5/SCK1/ SCL1	NC
136	P40/TMCIO/TxD2/ IrTxD	P40/TMCIO/TxD2/ IrTxD	P40/TMCIO/TxD2/ IrTxD	NC
137	P41/TMO0/RxD2/ IrRxD	P41/TMO0/RxD2/ IrRxD	P41/TMO0/RxD2/ IrRxD	NC
138 (N)	P42/TMRI0/SCK2/ SDA1	P42/TMRI0/SCK2/ SDA1	P42/TMRI0/SCK2/ SDA1	NC
139	VSS	VSS	VSS	VS
140	X1	X1	X1	NC
141	X2	X2	X2	NC
142	RES0	RES0	RES0	NC
143	XTAL	XTAL	XTAL	XT
144	EXTAL	EXTAL	EXTAL	EX

Note: The (B) in Pin No. means the VCCB drive and the (N) in Pin No. means the NMOS pull/open-drain drive.

	VCL	9	13	Input	Power supply pin. Connect to VCC.
	VCCB	4	36	Input	The power supply for the port input/output buffer.
	VSS	15, 70, 71, 92	7, 42, 95, 111, 139	Input	Ground pin. Connect to the system power supply (0 V).
Clock	XTAL	2	143	Input	Pins for connection to crystal resonators. The EXTAL pin can input an external clock. See section 25, Clock Pulse Generator for typical connection diagrams.
	EXTAL	3	144	Input	
	ϕ	17	18	Output	Supplies the system clock to external devices.
	EXCL	17	18	Input	Input a 32.768 kHz external signal.
	X1	—	140	Input	Leave open.
	X2	—	141	Input	Leave open.
	Operating mode control	MD1	5	9	Input
MD0		6	10		
System control	$\overline{\text{RES}}$	1	8	Input	Reset pin. When this pin becomes low, a reset occurs.
	$\overline{\text{RESO}}$	100	142	Output	Outputs reset signal to external devices.
	$\overline{\text{STBY}}$	8	12	Input	When this pin is driven low, a hardware standby is made.

Data bus	D15 to D8	89 to 82	128 to 121	Input/output	Bidirectional data bus for up to 16-bit data.
	D7 to D0	57, 58, 68, 69, 80, 81, 90, 91	113 to 120	Input/output	Bidirectional data bus for up to 16-bit data.
Bus control	$\overline{\text{WAIT}}$	16	17	Input	Requests insertion of a wait state during a bus cycle when accessing external address space.
	$\overline{\text{RD}}$	22	21	Output	When this pin is low, it indicates that the external address space is busy.
	$\overline{\text{HWR}}$	19	20	Output	When this pin is low, it indicates that the external address space is busy. The upper half of the data bus is not valid.
	$\overline{\text{LWR}}$	25	24	Output	When this pin is low, it indicates that the external address space is busy. The lower half of the data bus is not valid.
	$\overline{\text{AS/IOS}}$	18	19	Output	When this pin is low, it indicates that the address output on the address bus is not valid.
Interrupt signals	NMI	7	11	Input	Input pin for a nonmaskable interrupt request.
	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ7}}$	23 to 25, 97 to 99, 34, 35	22 to 24, 133 to 135, 84, 85	Input	These pins request a maskable interrupt.

	FTIB	29	81	Input	The input capture B input pins.
	FTIC	32	82	Input	The input capture C input pins.
	FTID	33	83	Input	The input capture D input pins.
8-bit timer (TMR_0, TMR_1, TMR_X)	TMO0	50	137	Output	The waveform output pins for compare function.
	TMO1	53	3		
	TMOX	35	85		
	TMC10	49	136	Input	Input pins for the external counters.
	TMC11	52	2		
	TMRI0	51	138	Input	The counter reset input pins.
	TMRI1	54	4		
8-bit timer (TMR_X, TMR_Y)	TMIX	26	78	Input	The counter event input and reset input pins.
	TMIY	28	80		
8-bit PWM timer (PWM)	PW15 to PW0	60 to 67, 72 to 79	96 to 110, 112	Output	PWM timer pulse output pins.
14-bit PWM timer (PWMX)	PWX0	55	5	Output	PWM D/A pulse output pins.
	PWX1	56	6		
Serial communi- cation interface (SCI_0, SCI_1, SCI_2)	TxD0	14	16	Output	Transmit data output pins.
	TxD1	97	133		
	TxD2	49	136		
	RxD0	13	15	Input	Receive data input pins.
	RxD1	98	134		
	RxD2	50	137		
	SCK0	12	14		
SCK1	99	135	Input/ Output	Clock input/output pins. The output type is NMOS pu	
SCK2	51	138			
SCI with IrDA (SCI_2)	IrTxD	49	136	Output	Input and output pins for data for IrDA use.
	IrRxD	50	137	Input	

Host interface (XBS)	HDB7 to HDB0	89 to 82	128 to 121	Input/Output	Bidirectional 8-bit bus for access to XBS.
	$\overline{CS1}$, $\overline{CS2}$, $\overline{ECS2}$, $\overline{CS3}$, $\overline{CS4}$	18, 94, 25, 81, 80	19, 130, 24, 118, 117	Input	Input pins for selecting XBS 0 to 4. The $\overline{CS2}$ or $\overline{ECS2}$ input is selected with the system control register.
	\overline{IOR}	22	21	Input	Input pin that enables reading from XBS.
	\overline{IOW}	19	20	Input	Input pin that enables writing to XBS.
	HA0	93	129	Input	Input pin that indicates whether access is a data access or command access.
	GA20	94	130	Output	A20 gate control signal output.
	HIRQ11 HIRQ1 HIRQ12 HIRQ3 HIRQ4	52 53 54 91 90	2 3 4 120 119	Output	Output pins for interrupt request to host.
	HIFSD	95	131	Input	Control input pin used to place input/output pins in the high-impedance cutoff state.
	Host interface (LPC)	LAD3 to LAD0	85 to 82	124 to 121	Input/Output
\overline{LFRAME}		86	125	Input	Input pin that indicates the start of an LPC cycle or forced termination of an abnormal LPC cycle.
\overline{LRESET}		87	126	Input	Input pin that indicates an L2 cache error.
LCLK		88	127	Input	The LPC clock input pin.

	GA20	94	130	Input/ Output	A20 gate control signal output Output state monitoring input possible.
	$\overline{\text{CLKRUN}}$	95	131	Input/ Output	Input/output pin that requests of LCLK operation when LCLK stopped.
	$\overline{\text{LPCPD}}$	96	132	Input	Input pin that controls LPC m shutdown.
Keyboard buffer controller	$\overline{\text{KIN0}}$ to $\overline{\text{KIN15}}$	26 to 29, 32 to 35, 48, 47, 31, 30, 21, 20, 11, 10	78 to 85, 41 to 37, 35 to 33	Input	Matrix keyboard input pins. $\overline{\text{K}}$ $\overline{\text{KIN15}}$ are used as key-scan P10 to P17 and P20 to P27 a key-scan outputs. This allows maximum 16-output \times 16-inp matrix to be configured.
	$\overline{\text{WUE0}}$ to $\overline{\text{WUE7}}$	91, 90, 81, 80, 69, 68, 58, 57	120 to 113	Input	Wakeup event input pins. Th allow the same kind of wakeu wakeup from various sources
A/D converter	AN7 to AN0	45 to 38	68 to 75	Input	Analog input pins.
	CIN0 to CIN15	26 to 29, 32 to 35, 48, 47, 31, 30, 21, 20, 11, 10	78 to 85, 41 to 37, 35 to 33	Input	A/D conversion input pins, bu they are also used as digital input/output pins, accuracy w
	$\overline{\text{ADTRG}}$	25	24	Input	Pin for input of an external tri start A/D conversion.
D/A converter	DA0	44	74	Output	Analog output pins.
	DA1	45	75		

	AVref	36	77	Input	The reference power supply for the A/D converter and D/A converter. When the A/D and D/A converters are not used, this pin should be connected to the system power supply.
	AVSS	46	67	Input	The ground pin for the A/D converter and D/A converter. This pin should be connected to the system power supply (0 V).
Timer connection	VSYNCI	28	80	Input	Timer connection synchronous input pins.
	HSYNCI	52	2		
	CSYNCI	54	4		
	VFBACKI	29	81		
	HFBACKI	26	78		
	VSYNCO	27	79	Output	Timer connection synchronous output pins.
	HSYNCO	53	3		
	CLAMPO	32	82		
	CBLANK	60	96		
I ² C bus interface (IIC)	SCL0	12	14	Input/	I ² C clock I/O pins. The output type is NMOS open-drain output.
	SCL1	99	135	Output	
	SDA0	16	17	Input/	
	SDA1	51	138	Output	
I/O ports	P17 to P10	72 to 79	104 to 110, 112	Input/ Output	Eight input/output pins.
	P27 to P20	60 to 67	96 to 103	Input/ Output	Eight input/output pins.
	P37 to P30	89 to 82	128 to 121	Input/ Output	Eight input/output pins.
	P47 to P40	56 to 49	6 to 2, 138 to 136	Input/ Output	Eight input/output pins. (The output type of P42 is N-channel open-drain pull-up.)

P77 to P70	45 to 38	75 to 68	Input	Eight input pins.
P86 to P80	99 to 93	135 to 129	Input/Output	Seven input/output pins. (The output type of P86 is NMOS pull.)
P97 to P90	16 to 19 22 to 25	17 to 24	Input/Output	Eight input/output pins. (The output type of P97 is NMOS pull.)
PA7 to PA0	10, 11, 20, 21, 30, 31, 47, 48	33 to 35, 37 to 41	Input/Output	Eight input/output pins.
PB7 to PB0	57, 58, 68, 69, 80, 81, 90, 91	113 to 120	Input/Output	Eight input/output pins.
PC7 to PC0	—	87 to 94	Input/Output	Eight input/output pins.
PD7 to PD0	—	59 to 66	Input/Output	Eight input/output pins.
PE7 to PE0	—	25 to 32	Input/Output	Eight input/output pins.
PF7 to PF0	—	43 to 50	Input/Output	Eight input/output pins.
PG7 to PG0	—	51 to 58	Input/Output	Eight input/output pins. (The output type of PG7 to PG0 is NMOS push-pull.)

2.1 Features

- Upward-compatibility with H8/300 and H8/300H CPUs
Can execute H8/300 CPU and H8/300H CPU object programs
- General-register architecture
Sixteen 16-bit general registers also usable as sixteen 8-bit registers or eight 32-bit registers
- Sixty-five basic instructions
8/16/32-bit arithmetic and logic instructions
Multiply and divide instructions
Powerful bit-manipulation instructions
- Eight addressing modes
Register direct [Rn]
Register indirect [@ERn]
Register indirect with displacement [@(d:16,ERn) or @(d:32,ERn)]
Register indirect with post-increment or pre-decrement [@ERn+ or @-ERn]
Absolute address [@aa:8, @aa:16, @aa:24, or @aa:32]
Immediate [#xx:8, #xx:16, or #xx:32]
Program-counter relative [@(d:8,PC) or @(d:16,PC)]
Memory indirect [@@aa:8]
- 16-Mbyte address space
Program: 16 Mbytes
Data: 16 Mbytes
- High-speed operation
All frequently-used instructions are executed in one or two states
8/16/32-bit register-register add/subtract: 1 state
8 × 8-bit register-register multiply: 12 states (MULXU.B), 13 states (MULXS.B)
16 ÷ 8-bit register-register divide: 12 states (DIVXU.B)

2.1.1 Differences between H8S/2600 CPU and H8S/2000 CPU

The differences between the H8S/2600 CPU and the H8S/2000 CPU are as shown below.

- Register configuration

The MAC register is supported only by the H8S/2600 CPU.

- Basic instructions

The four instructions MAC, CLRMAC, LDMAC, and STMAC are supported only by the H8S/2600 CPU.

- The number of execution states of the MULXU and MULXS instructions

Instruction	Mnemonic	Execution States	
		H8S/2600	H8S/2000
MULXU	MULXU.B Rs, Rd	3	12
	MULXU.W Rs, ERd	4	20
MULXS	MULXS.B Rs, Rd	4	13
	MULXS.W Rs, ERd	5	21

In addition, there are differences in address space, CCR and EXR register functions, power modes, etc., depending on the model.

- Advanced mode supports a maximum 16-Mbyte address space.
- Enhanced addressing
The addressing modes have been enhanced to make effective use of the 16-Mbyte space.
- Enhanced instructions
Addressing modes of bit-manipulation instructions have been enhanced.
Signed multiply and divide instructions have been added.
Two-bit shift and two-bit rotate instructions have been added.
Instructions for saving and restoring multiple registers have been added.
A test and set instruction has been added.
- Higher speed
Basic instructions are executed twice as fast.

2.1.3 Differences from H8/300H CPU

In comparison to the H8/300H CPU, the H8S/2000 CPU has the following enhancements:

- Additional control register
One 8-bit control register has been added.
- Enhanced instructions
Addressing modes of bit-manipulation instructions have been enhanced.
Two-bit shift and two-bit rotate instructions have been added.
Instructions for saving and restoring multiple registers have been added.
A test and set instruction has been added.
- Higher speed
Basic instructions are executed twice as fast.

The exception vector table and stack have the same structure as in the H8/300 CPU in normal mode.

- Address space

Linear access to a maximum address space of 64 kbytes is possible.

- Extended registers (En)

The extended registers (E0 to E7) can be used as 16-bit registers, or as the upper 16-bit segments of 32-bit registers.

When extended register En is used as a 16-bit register it can contain any value, even if the corresponding general register (Rn) is used as an address register. (If general register Rn is referenced in the register indirect addressing mode with pre-decrement (@-Rn) or post-increment (@Rn+) and a carry or borrow occurs, the value in the corresponding extended register (En) will be affected.)

- Instruction set

All instructions and addressing modes can be used. Only the lower 16 bits of effective addresses (EA) are valid.

- Exception vector table and memory indirect branch addresses

In normal mode, the top area starting at H'0000 is allocated to the exception vector table. A branch address is stored per 16 bits. The exception vector table in normal mode is shown in figure 2.1. For details on the exception vector table, see section 4, Exception Handling.

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory location that contains a branch address. In normal mode, the operand is a 16-bit (word) operation providing a 16-bit branch address. Branch addresses can be stored in the top area from H'0000 to H'00FF. Note that this area is also used for the exception vector table.

- Stack structure

In normal mode, when the program counter (PC) is pushed onto the stack in a subroutine call, the PC and condition-code register (CCR) are pushed onto the stack. During exception handling, they are stored as shown in figure 2.2. The extended control register (EXR) is not pushed onto the stack. For details, see section 4, Exception Handling.

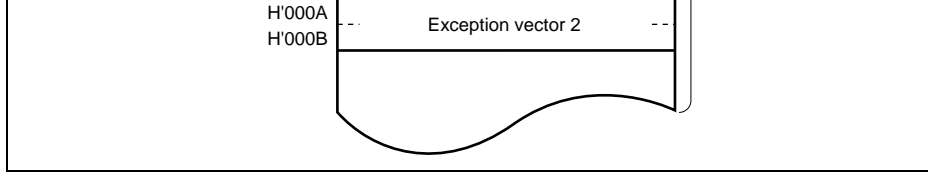


Figure 2.1 Exception Vector Table (Normal Mode)

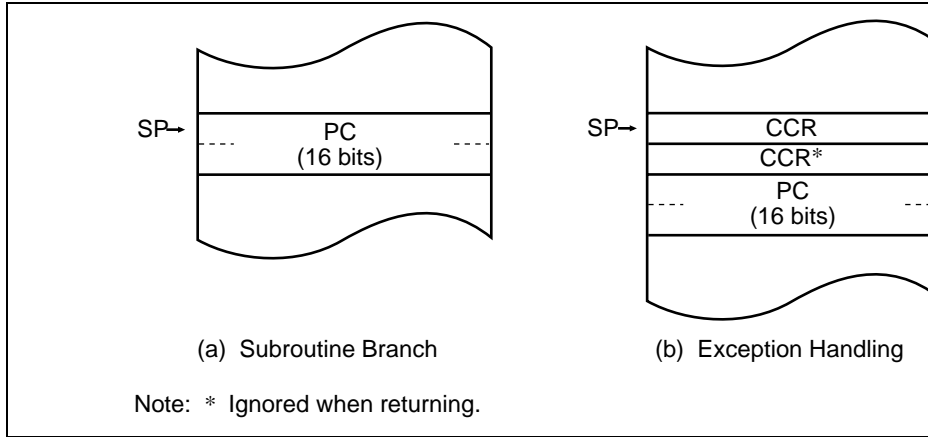


Figure 2.2 Stack Structure in Normal Mode

2.2.2 Advanced Mode

- Address space
Linear access to a maximum address space of 16 Mbytes is possible.
- Extended registers (En)
The extended registers (E0 to E7) can be used as 16-bit registers. They can also be used as upper 16-bit segments of 32-bit registers or address registers.

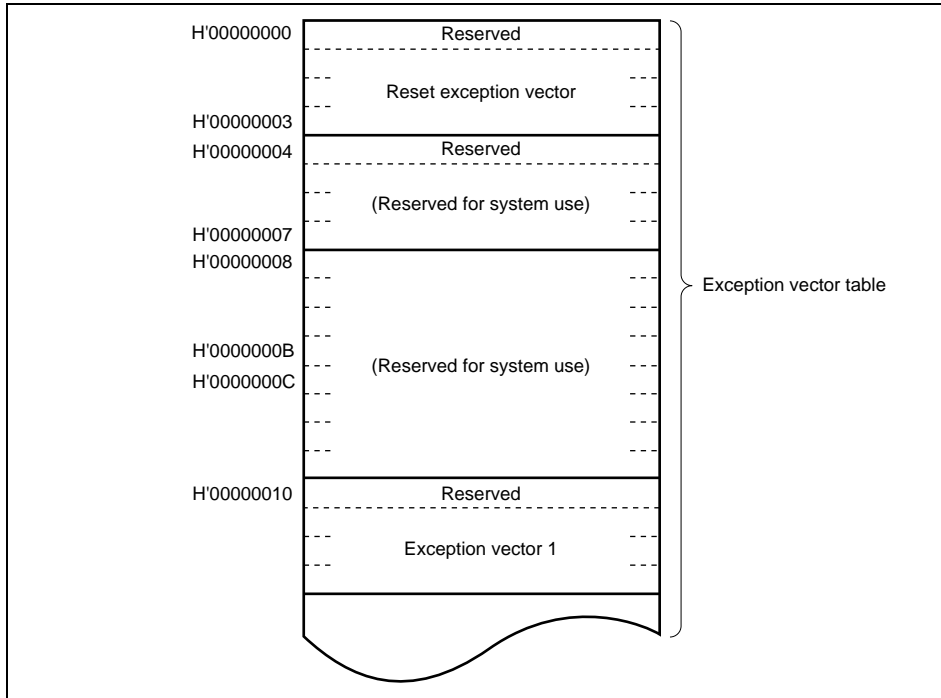


Figure 2.3 Exception Vector Table (Advanced Mode)

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory location that contains a branch address. In advanced mode, the operand is a 32-bit longword, providing a 32-bit branch address. The upper 8 bits of these 32 bits are a reserved area regarded as H'00. Branch addresses can be stored in the area from H'00000000 to H'00000010. Note that the top area of this range is also used for the exception vector table.

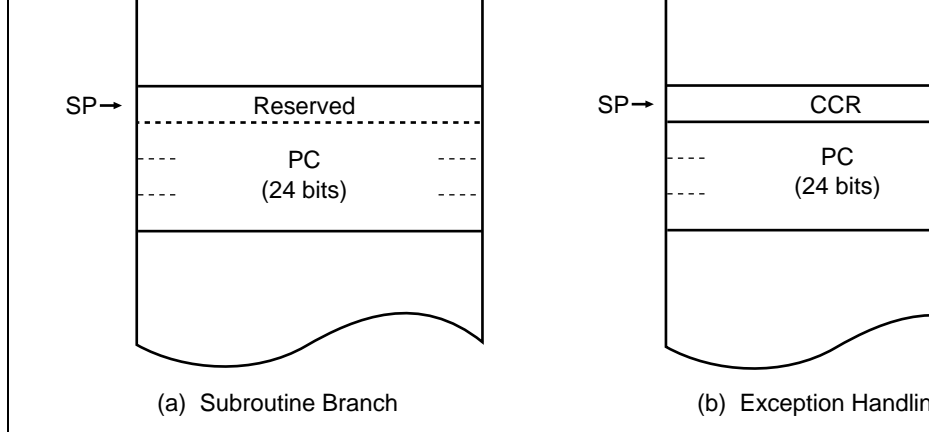


Figure 2.4 Stack Structure in Advanced Mode

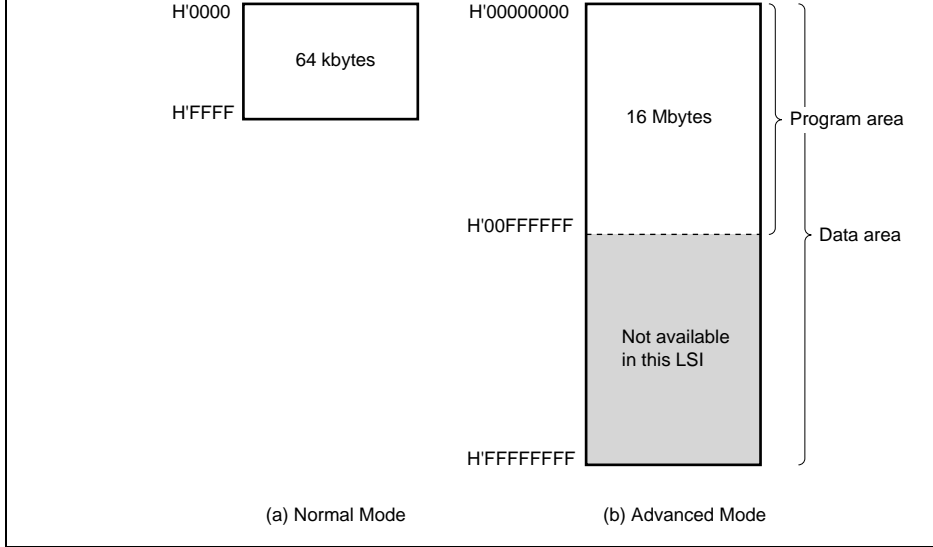
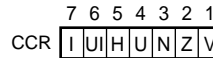
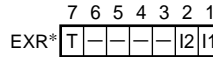
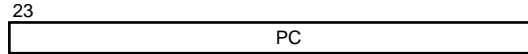


Figure 2.5 Memory Map

ER0	E0	R0H	R0L
ER1	E1	R1H	R1L
ER2	E2	R2H	R2L
ER3	E3	R3H	R3L
ER4	E4	R4H	R4L
ER5	E5	R5H	R5L
ER6	E6	R6H	R6L
ER7 (SP)	E7	R7H	R7L

Control Registers



Legend:

SP	: Stack pointer	H	: Half-carry flag
PC	: Program counter	U	: User bit
EXR	: Extended control register	N	: Negative flag
T	: Trace bit	Z	: Zero flag
I2 to I0	: Interrupt mask bits	V	: Overflow flag
CCR	: Condition-code register	C	: Carry flag
I	: Interrupt mask bit		
UI	: User bit or interrupt mask bit		

Note: * Does not affect operation in this LSI.

Figure 2.6 CPU Internal Registers

When the general registers are used as 16-bit registers, the ER registers are divided into general registers designated by the letters E (E0 to E7) and R (R0 to R7). These registers are functionally equivalent, providing a maximum sixteen 16-bit registers. The E registers are also referred to as extended registers.

When the general registers are used as 8-bit registers, the R registers are divided into 8-bit registers designated by the letters RH (R0H to R7H) and RL (R0L to R7L). These registers are functionally equivalent, providing a maximum sixteen 8-bit registers.

The usage of each register can be selected independently.

General register ER7 has the function of the stack pointer (SP) in addition to its general register function, and is used implicitly in exception handling and subroutine calls. Figure 2.8 shows the stack.

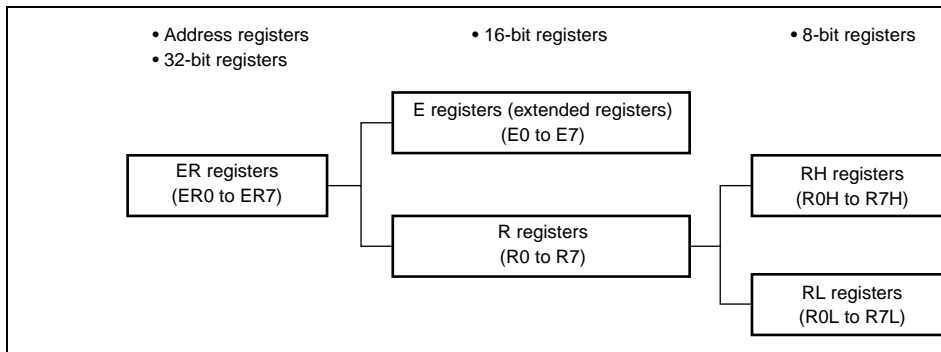


Figure 2.7 Usage of General Registers



Figure 2.8 Stack

2.4.2 Program Counter (PC)

This 24-bit counter indicates the address of the next instruction the CPU will execute. The address of all CPU instructions is 2 bytes (one word), so the least significant PC bit is ignored (when the next instruction is fetched for read, the least significant PC bit is regarded as 0.)

2.4.3 Extended Control Register (EXR)

EXR does not affect operation in this LSI.

Bit	Bit Name	Initial Value	R/W	Description
7	T	0	R/W	Trace Bit Does not affect operation in this LSI.
6 to 3	—	All 1	R	Reserved These bits are always read as 1.
2 to 0	I2 I1 I0	All 1	R/W	Interrupt Mask Bits 2 to 0 Do not affect operation in this LSI.

2.4.4 Condition-Code Register (CCR)

This 8-bit register contains internal CPU status information, including an interrupt mask (I), half-carry (H), negative (N), zero (Z), overflow (V), and carry (C) flags. Operations can be performed on the CCR bits by the LDC, STC, ANDC, ORC, and XORC instructions. The H, N, Z, V, and C flags are used as branching conditions for conditional branch (Bcc) instructions.

5	H	Undefined	R/W	<p>Half-Carry Flag</p> <p>When the ADD.B, ADDX.B, SUB.B, SUBX.B, NEG.B instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and cleared to 0 otherwise.</p> <p>When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 11, and cleared to 0 otherwise.</p> <p>When the ADD.L, SUB.L, CMP.L, or NEG.L instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 27, and cleared to 0 otherwise.</p>
4	U	Undefined	R/W	<p>User Bit</p> <p>Can be written to and read from by software using the LDC, STC, ANDC, ORC, and XORC instructions.</p>
3	N	Undefined	R/W	<p>Negative Flag</p> <p>Stores the value of the most significant bit of the operand sign bit.</p>
2	Z	Undefined	R/W	<p>Zero Flag</p> <p>Set to 1 to indicate zero data, and cleared to 0 to indicate non-zero data.</p>
1	V	Undefined	R/W	<p>Overflow Flag</p> <p>Set to 1 when an arithmetic overflow occurs, and cleared to 0 otherwise.</p>
0	C	Undefined	R/W	<p>Carry Flag</p> <p>Set to 1 when a carry occurs, and cleared to 0 otherwise.</p> <p>Used by</p> <ul style="list-style-type: none"> • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a carry <p>The carry flag is also used as a bit accumulator for carry manipulation instructions.</p>

2.5 Data Formats

The H8S/2000 CPU can process 1-bit, 4-bit BCD, 8-bit (byte), 16-bit (word), and 32-bit (longword) data. Bit-manipulation instructions operate on 1-bit data by accessing bit number 0 to 7 (bit 0, bit 1, ..., bit 7) of byte operand data. The DAA and DAS decimal-adjust instructions treat byte operand data as 4-bit BCD digits of 4-bit BCD data.

2.5.1 General Register Data Formats

Figure 2.9 shows the data formats of general registers.

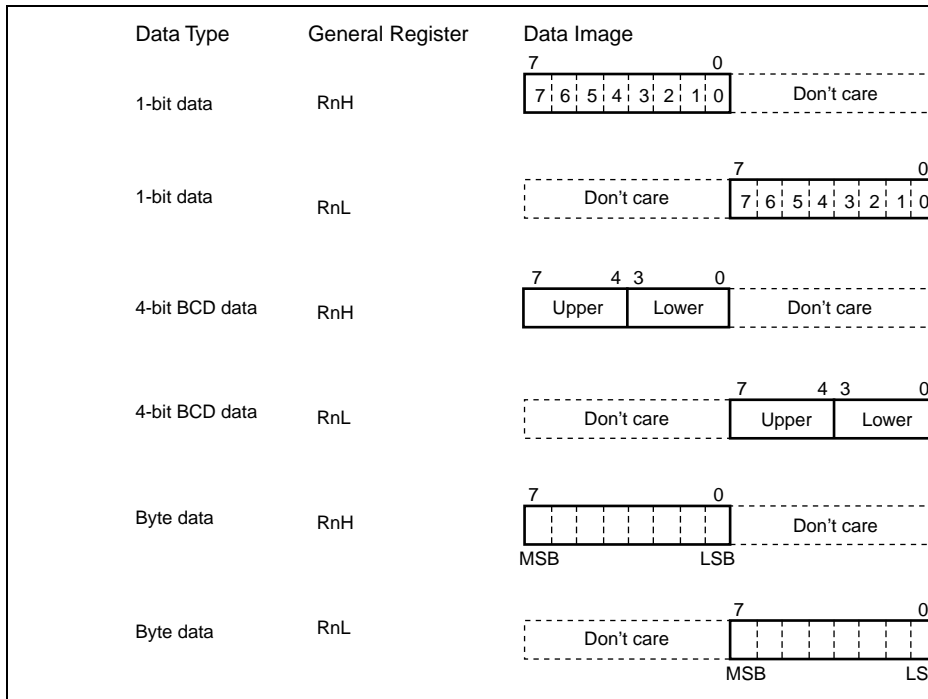


Figure 2.9 General Register Data Formats (1)

size or longword size.

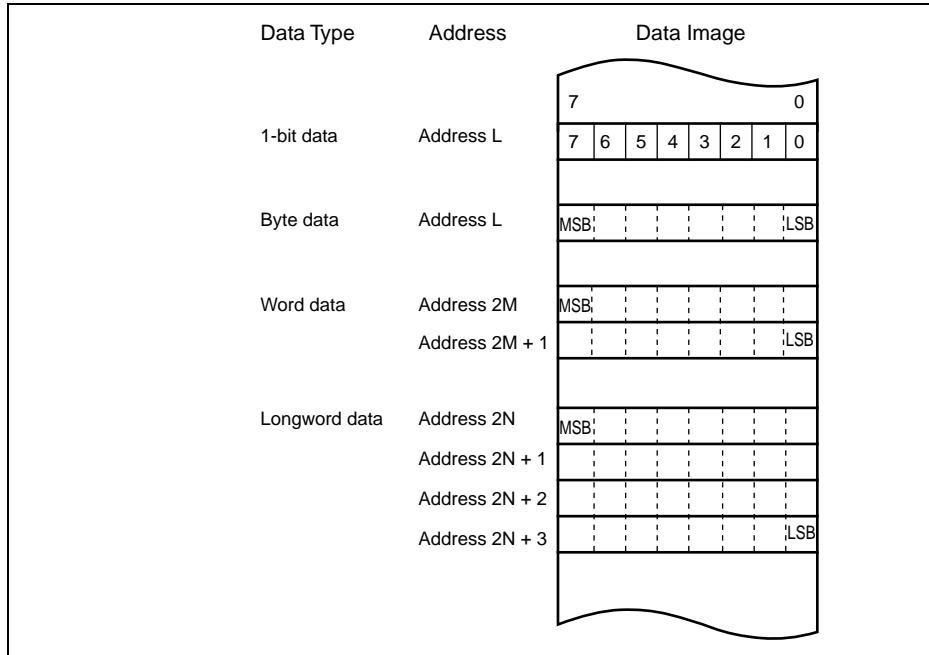


Figure 2.10 Memory Data Formats

	POP ^{*1} , PUSH ^{*1}	W/L
	LDM ^{*5} , STM ^{*5}	L
	MOVFP ^{*3} , MOVTP ^{*3}	B
Arithmetic operations	ADD, SUB, CMP, NEG	B/W/L
	ADDX, SUBX, DAA, DAS	B
	INC, DEC	B/W/L
	ADDS, SUBS	L
	MULXU, DIVXU, MULXS, DIVXS	B/W
	EXTU, EXTS	W/L
	TAS ^{*4}	B
Logic operations	AND, OR, XOR, NOT	B/W/L
Shift	SHAL, SHAR, SHLL, SHLR, ROTL, ROTR, ROTXL, ROTXR	B/W/L
Bit manipulation	BSET, BCLR, BNOT, BTST, BLD, BILD, BST, BIST, BAND, BIAND, BOR, BIOR, BXOR, BIXOR	B
Branch	Bcc ^{*2} , JMP, BSR, JSR, RTS	—
System control	TRAPA, RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP	—
Block data transfer	EEPMOV	—

Legend: B: Byte size
W: Word size
L: Longword size.

- Notes:
1. POP.W Rn and PUSH.W Rn are identical to MOV.W @SP+, Rn and MOV.W @-SP. POP.L ERn and PUSH.L ERn are identical to MOV.L @SP+, ERn and MOV.L @-SP, ERn.
 2. Bcc is the general name for conditional branch instructions.
 3. Cannot be used in this LSI.
 4. When using the TAS instruction, use registers ER0, ER1, ER4, and ER5.
 5. ER7 is not used as the register that can be saved (STM)/restored (LDM) when using the STM/LDM instruction, because ER7 is the stack pointer.

Rs	General register (source)*
Rn	General register*
ERn	General register (32-bit register)
(EAd)	Destination operand
(EAs)	Source operand
EXR	Extended control register
CCR	Condition-code register
N	N (negative) flag in CCR
Z	Z (zero) flag in CCR
V	V (overflow) flag in CCR
C	C (carry) flag in CCR
PC	Program counter
SP	Stack pointer
#IMM	Immediate data
disp	Displacement
+	Addition
-	Subtraction
×	Multiplication
÷	Division
^	Logical AND
∨	Logical OR
⊕	Logical exclusive OR
→	Move
~	NOT (logical complement)
:8/:16/:24/:32	8-, 16-, 24-, or 32-bit length

Note: * General registers include 8-bit registers (R0H to R7H, R0L to R7L), 16-bit registers (R8 to R7, E0 to E7), and 32-bit registers (ER0 to ER7).

POP	W/L	@SP+ → Rn Pops a general register from the stack. POP.W Rn is identical to MOV.W Rn, @SP+. POP.L ERn is identical to MOV.L @SP+, ERn
PUSH	W/L	Rn → @-SP Pushes a general register onto the stack. PUSH.W Rn is identical to MOV.W Rn, @-SP. PUSH.L ERn is identical to MOV.L ERn, @-SP.
LDM ^{*2}	L	@SP+ → Rn (register list) Pops two or more general registers from the stack.
STM ^{*2}	L	Rn (register list) → @-SP Pushes two or more general registers onto the stack.

- Notes:
1. Size refers to the operand size.
B: Byte
W: Word
L: Longword
 2. ER7 is not used as the register that can be saved (STM)/restored (LDM) when using the STM/LDM instruction, because ER7 is the stack pointer.

SUBX		Performs addition or subtraction with carry on data in two general registers, or on immediate data and data in a general register.
INC	B/W/L	$Rd \pm 1 \rightarrow Rd, Rd \pm 2 \rightarrow Rd$
DEC		Adds or subtracts the value 1 or 2 to or from data in a general register. (Only the value 1 can be added to or subtracted from byte operands.)
ADDS	L	$Rd \pm 1 \rightarrow Rd, Rd \pm 2 \rightarrow Rd, Rd \pm 4 \rightarrow Rd$
SUBS		Adds or subtracts the value 1, 2, or 4 to or from data in a 32-bit register.
DAA	B	Rd (decimal adjust) $\rightarrow Rd$
DAS		Decimal-adjusts an addition or subtraction result in a general register, referring to CCR to produce 4-bit BCD data.
MULXU	B/W	$Rd \times Rs \rightarrow Rd$ Performs unsigned multiplication on data in two general registers. 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits.
MULXS	B/W	$Rd \times Rs \rightarrow Rd$ Performs signed multiplication on data in two general registers. 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits.
DIVXU	B/W	$Rd \div Rs \rightarrow Rd$ Performs unsigned division on data in two general registers. 32 bits \div 8 bits \rightarrow 8-bit quotient and 8-bit remainder or 32 bits \div 16 bits \rightarrow 16-bit quotient and 16-bit remainder.

Note: * Size refers to the operand size.

B: Byte

W: Word

L: Longword

Compares data in a general register with data in another general register or with immediate data, and sets the CCR bits according to the result.

NEG	B/W/L	0 – Rd → Rd Takes the two's complement (arithmetic complement) of data in the specified general register.
EXTU	W/L	Rd (zero extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by padding with zeros on the left.
EXTS	W/L	Rd (sign extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by extending the sign bit.
TAS ^{*2}	B	@ERd – 0, 1 → (<bit 7> of @ERd) Tests memory contents, and sets the most significant bit (bit 7) of the specified register.

Notes: 1. Size refers to the operand size.

B: Byte

W: Word

L: Longword

2. When using the TAS instruction, use registers ER0, ER1, ER4 and ER5.

XOR	B/W/L	$Rd \oplus Rs \rightarrow Rd, Rd \oplus \#IMM \rightarrow Rd$ Performs a logical exclusive OR operation on a general register or immediate data and another general register or immediate data.
NOT	B/W/L	$\sim Rd \rightarrow Rd$ Takes the one's complement (logical complement) of data in a general register.

Note: * Size refers to the operand size.

B: Byte

W: Word

L: Longword

Table 2.6 Shift Instructions

Instruction	Size*	Function
SHAL	B/W/L	$Rd \text{ (shift)} \rightarrow Rd$
SHAR		Performs an arithmetic shift on data in a general register. 1-bit or all bits shift is possible.
SHLL	B/W/L	$Rd \text{ (shift)} \rightarrow Rd$
SHLR		Performs a logical shift on data in a general register. 1-bit or all bits shift is possible.
ROTL	B/W/L	$Rd \text{ (rotate)} \rightarrow Rd$
ROTR		Rotates data in a general register. 1-bit or 2 bit rotation is possible.
ROTXL	B/W/L	$Rd \text{ (rotate)} \rightarrow Rd$
ROTXR		Rotates data including the carry flag in a general register. 1-bit or all bits rotation is possible.

Note: * Size refers to the operand size.

B: Byte

W: Word

L: Longword

Clears a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.

BNOT	B	$\sim (\text{<bit-No.> of <EAd>}) \rightarrow (\text{<bit-No.> of <EAd>})$ Inverts a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.
BTST	B	$\sim (\text{<bit-No.> of <EAd>}) \rightarrow Z$ Tests a specified bit in a general register or memory operand and clears the Z flag accordingly. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.
BAND	B	$C \wedge (\text{<bit-No.> of <EAd>}) \rightarrow C$ Logically ANDs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIAND	B	$C \wedge [\sim (\text{<bit-No.> of <EAd>})] \rightarrow C$ Logically ANDs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.
BOR	B	$C \vee (\text{<bit-No.> of <EAd>}) \rightarrow C$ Logically ORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIOR	B	$C \vee [\sim (\text{<bit-No.> of <EAd>})] \rightarrow C$ Logically ORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.

Note: * Size refers to the operand size.

B: Byte

in a general register or memory operand and stores the result in the carry flag.

The bit number is specified by 3-bit immediate data.

BLD	B	(<bit-No.> of <EAd>) → C Transfers a specified bit in a general register or memory operand to the carry flag.
BILD	B	~ (<bit-No.> of <EAd>) → C Transfers the inverse of a specified bit in a general register or memory operand to the carry flag. The bit number is specified by 3-bit immediate data.
BST	B	C → (<bit-No.> of <EAd>) Transfers the carry flag value to a specified bit in a general register or memory operand.
BIST	B	~ C → (<bit-No.> of <EAd>) Transfers the inverse of the carry flag value to a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data.

Note: * Size refers to the operand size.

B: Byte

BHI	High	$C \vee Z = 0$
BLS	Low or same	$C \vee Z = 1$
BCC (BHS)	Carry clear (high or same)	$C = 0$
BCS (BLO)	Carry set (low)	$C = 1$
BNE	Not equal	$Z = 0$
BEQ	Equal	$Z = 1$
BVC	Overflow clear	$V = 0$
BVS	Overflow set	$V = 1$
BPL	Plus	$N = 0$
BMI	Minus	$N = 1$
BGE	Greater or equal	$N \oplus V = 0$
BLT	Less than	$N \oplus V = 1$
BGT	Greater than	$Z \vee (N \oplus V) = 0$
BLE	Less or equal	$Z \vee (N \oplus V) = 1$

JMP	—	Branches unconditionally to a specified address.
BSR	—	Branches to a subroutine at a specified address
JSR	—	Branches to a subroutine at a specified address
RTS	—	Returns from a subroutine

EXR. Although CCR and EXR are 8-bit registers, word-size transfers are performed between them and memory. The upper 8 bits

STC	B/W	CCR \rightarrow (EAd), EXR \rightarrow (EAd) Transfers CCR or EXR contents to a general register or memory operand. Although CCR and EXR are 8-bit registers, word-size transfers are performed between them and memory. The upper 8 bits are valid.
ANDC	B	CCR \wedge #IMM \rightarrow CCR, EXR \wedge #IMM \rightarrow EXR Logically ANDs the CCR or EXR contents with immediate data.
ORC	B	CCR \vee #IMM \rightarrow CCR, EXR \vee #IMM \rightarrow EXR Logically ORs the CCR or EXR contents with immediate data.
XORC	B	CCR \oplus #IMM \rightarrow CCR, EXR \oplus #IMM \rightarrow EXR Logically exclusive-ORs the CCR or EXR contents with immediate data.
NOP	—	PC + 2 \rightarrow PC Only increments the program counter.

Note: * Size refers to the operand size.

B: Byte

W: Word

Repeat @ER5 + → @ER6+

R4-1 → R4

Until R4 = 0

else next;

Transfers a data block. Starting from the address set in ER5 data for the number of bytes set in R4L or R4 to the address set in ER6.

Execution of the next instruction begins as soon as the transfer is completed.

2.6.2 Basic Instruction Formats

The H8S/2000 CPU instructions consist of 2-byte (1-word) units. An instruction consists of an operation field (op), a register field (r), an effective address extension (EA), and a condition code (cc).

Figure 2.11 shows examples of instruction formats.

- **Operation field**
Indicates the function of the instruction, the addressing mode, and the operation to be performed on the operand. The operation field always includes the first four bits of the instruction. Some instructions have two operation fields.
- **Register field**
Specifies a general register. Address registers are specified by 3 bits, and data registers by 4 bits. Some instructions have two register fields, and some have no register fields.
- **Effective address extension**
8, 16, or 32 bits specifying immediate data, an absolute address, or a displacement.
- **Condition field**
Specifies the branching condition of Bcc instructions.

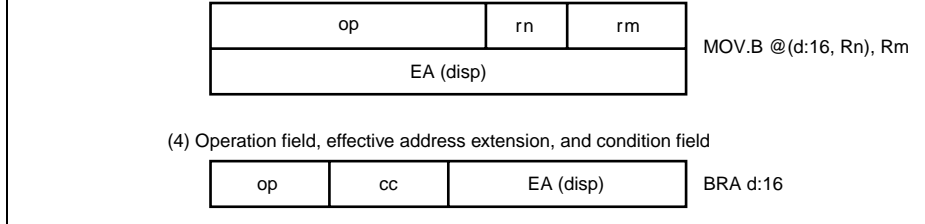


Figure 2.11 Instruction Formats (Examples)

2.7 Addressing Modes and Effective Address Calculation

The H8S/2000 CPU supports the eight addressing modes listed in table 2.11. Each instruction uses a subset of these addressing modes.

Arithmetic and logic operations instructions can use the register direct and immediate addressing modes. Data transfer instructions can use all addressing modes except program-counter indirect and memory indirect. Bit manipulation instructions can use register direct, register indirect, and absolute addressing mode to specify an operand, and register direct (BSET, BCLR, BIT, BTST instructions) or immediate (3-bit) addressing mode to specify a bit number in the

5	Absolute address	@aa:8/@aa:16/@aa:24/@aa:32
6	Immediate	#xx:8/#xx:16/#xx:32
7	Program-counter relative	@(d:8,PC)/@(d:16,PC)
8	Memory indirect	@@aa:8

2.7.1 Register Direct—Rn

The register field of the instruction code specifies an 8-, 16-, or 32-bit general register which contains the operand. R0H to R7H and R0L to R7L can be specified as 8-bit registers. R0 to R7 and E0 to E7 can be specified as 16-bit registers. ER0 to ER7 can be specified as 32-bit registers.

2.7.2 Register Indirect—@ERn

The register field of the instruction code specifies an address register (ERn) which contains the address of a memory operand. If the address is a program instruction address, the lower 8 bits are valid and the upper 8 bits are all assumed to be 0 (H'00).

2.7.3 Register Indirect with Displacement—@(d:16, ERn) or @(d:32, ERn)

A 16-bit or 32-bit displacement contained in the instruction code is added to an address register (ERn) specified by the register field of the instruction, and the sum gives the address of the memory operand. A 16-bit displacement is sign-extended when added.

2.7.4 Register Indirect with Post-Increment or Pre-Decrement—@ERn+ or @ERn-

Register Indirect with Post-Increment—@ERn+: The register field of the instruction code specifies an address register (ERn) which contains the address of a memory operand. After the operand is accessed, 1, 2, or 4 is added to the address register contents and the sum is stored in the address register. The value added is 1 for byte access, 2 for word access, and 4 for longword access. For word or longword transfer instructions, the register value should be even.

The instruction code contains the absolute address of a memory operand. The absolute address may be 8 bits long (@aa:8), 16 bits long (@aa:16), 24 bits long (@aa:24), or 32 bits long (@aa:32). Table 2.12 indicates the accessible absolute address ranges.

To access data, the absolute address should be 8 bits (@aa:8), 16 bits (@aa:16), or 32 bits (@aa:32) long. For an 8-bit absolute address, the upper 24 bits are all assumed to be 1. For a 16-bit absolute address, the upper 16 bits are a sign extension. For a 32-bit absolute address, the entire address space is accessed.

A 24-bit absolute address (@aa:24) indicates the address of a program instruction. The upper 8 bits are all assumed to be 0 (H'00).

Table 2.12 Absolute Address Access Ranges

Absolute Address	Normal Mode	Advanced Mode
Data address	8 bits (@aa:8)	H'FFF00 to H'FFFF
	16 bits (@aa:16)	H'0000 to H'FFFF
	32 bits (@aa:32)	H'000000 to H'FFFFFF
Program instruction address	24 bits (@aa:24)	

2.7.6 Immediate—#xx:8, #xx:16, or #xx:32

The 8-bit (#xx:8), 16-bit (#xx:16), or 32-bit (#xx:32) immediate data contained in an instruction code can be used directly as an operand.

The ADDS, SUBS, INC, and DEC instructions implicitly contain immediate data in their instruction codes. Some bit manipulation instructions contain 3-bit immediate data in their instruction code, specifying a bit number. The TRAPA instruction contains 2-bit immediate data in its instruction code, specifying a vector address.

2.7.8 Memory Indirect—@@aa:8

This mode can be used by the JMP and JSR instructions. The instruction code contains an absolute address specifying a memory operand which contains a branch address. The upper 8 bits of the 8-bit absolute address are all assumed to be 0, so the address range is 0 to 255 (H'00 to H'00FF in normal mode, H'000000 to H'0000FF in advanced mode).

In normal mode, the memory operand is a word operand and the branch address is 16 bits. In advanced mode, the memory operand is a longword operand, the first byte of which is assumed to be 0 (H'00). Note that the top area of the address range in which the branch address is specified is also used for the exception vector area. For further details, refer to section 4, Exception Handling.

If an odd address is specified in word or longword memory access, or as a branch address, the least significant bit is regarded as 0, causing data to be accessed or the instruction code to be fetched at the address preceding the specified address. (For further information, see section 2.7.1 Memory Data Formats.)

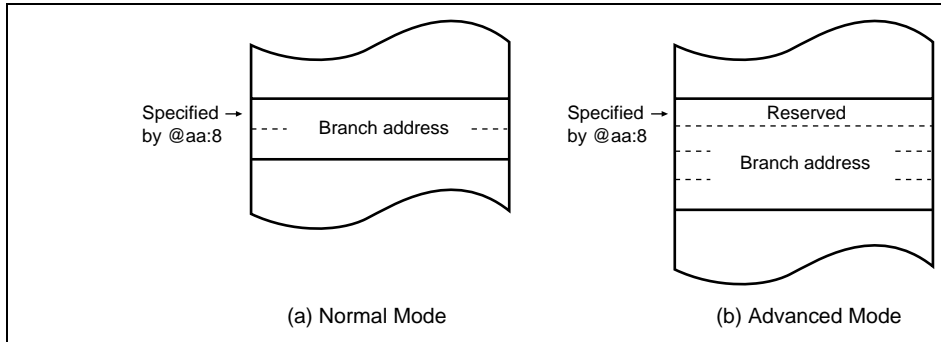
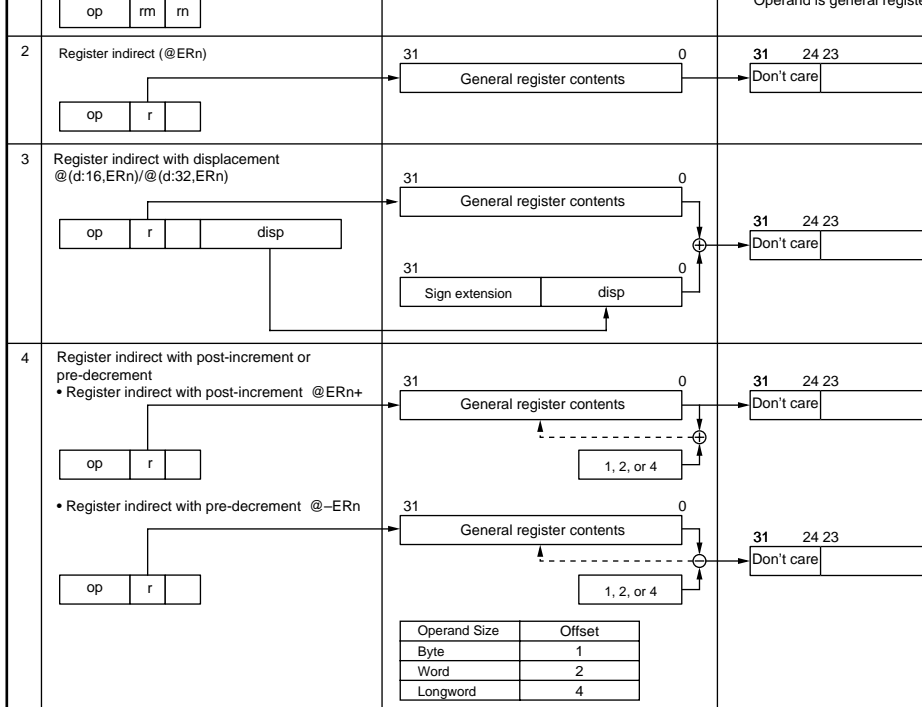
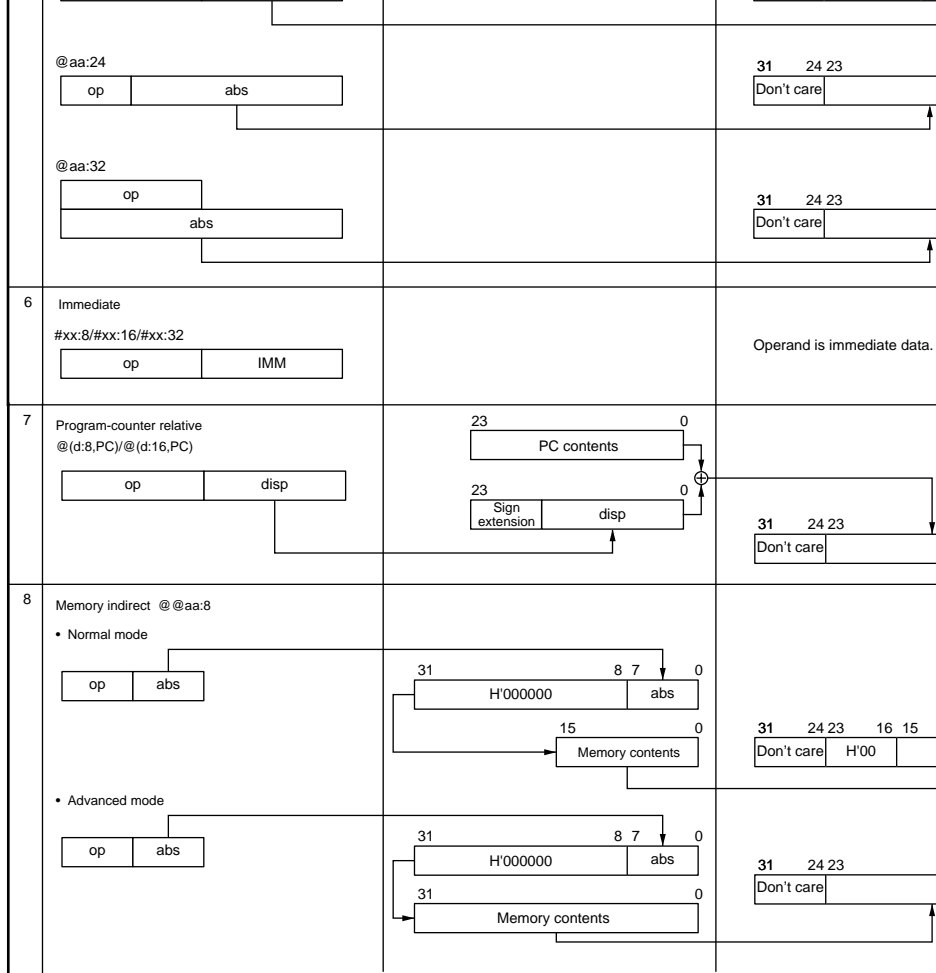


Figure 2.12 Branch Address Specification in Memory Indirect Addressing





RES input goes low, all current processing stops and the CPU enters the reset state. When the RES input changes from low to high, the CPU enters the program execution state. Reset exception handling starts when the RES input changes from low to high. For details, refer to section 4, Exception Handling.

The reset state can also be entered by a watchdog timer overflow.

- Exception-handling state

The exception-handling state is a transient state that occurs when the CPU alters the normal processing flow due to an exception source, such as, a reset, trace, interrupt, or trap. The CPU fetches a start address (vector) from the exception vector table and branches to that address. For further details, refer to section 4, Exception Handling.

- Program execution state

In this state the CPU executes program instructions in sequence.

- Bus-released state

In a product which has a bus master other than the CPU, such as a data transfer controller (DTC), the bus-released state occurs when the bus has been released in response to a request from a bus master other than the CPU. While the bus is released, the CPU performs no operations. For details, see section 6, Bus Controller (BSC).

- Program stop state

This is a power-down state in which the CPU stops operating. The program stop state occurs when a SLEEP instruction is executed or the CPU enters hardware standby mode. For details, refer to section 26, Power-Down Modes.

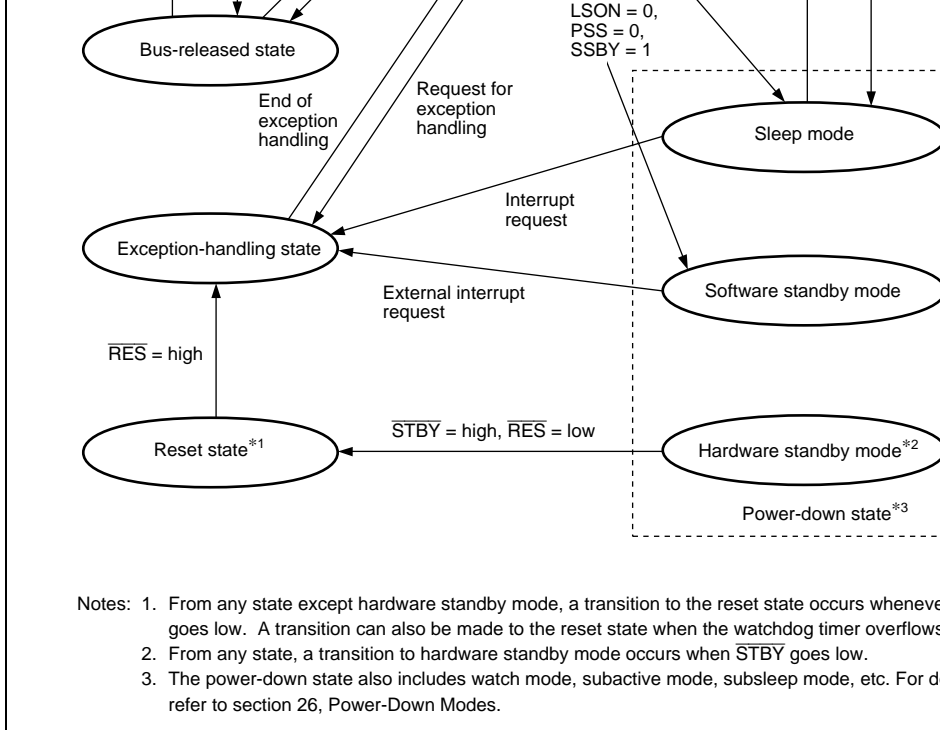


Figure 2.13 State Transitions

ER0, ER1, ER4 and ER5.

2.9.2 Note on STM/LDM Instruction Usage

ER7 is not used as the register that can be saved (STM)/restored (LDM) when using S instruction, because ER7 is the stack pointer. Two, three, or four registers can be saved by one STM/LDM instruction. The following ranges can be specified in the register list.

Two registers: ER0–ER1, ER2–ER3, or ER4–ER5

Three registers: ER0–ER2 or ER4–ER6

Four registers: ER0–ER3

The STM/LDM instruction including ER7 is not generated by the Renesas Technology H8/300 series C/C++ compilers.

2.9.3 Note on Bit Manipulation Instructions

The BSET, BCLR, BNOT, BST, and BIST instructions read data from the specified address byte units, manipulate the data of the target bit, and write data to the same address again. Special care is required when using these instructions in cases where a register's write-only bit is used or a bit is directly manipulated for a port, because this may rewrite a bit other than the bit to be manipulated.

Example: The BCLR instruction is executed for DDR in port 4.

P47 and P46 are input pins, with a low-level signal input at P47 and a high-level signal at P46. P45 to P40 are output pins and output low-level signals. The following shows an example in which P40 is set to be an input pin with the BCLR instruction.

BCLR instruction executed:

```
BCLR #0, @P4DDR
```

The BCLR instruction is executed for DDR in port 4.

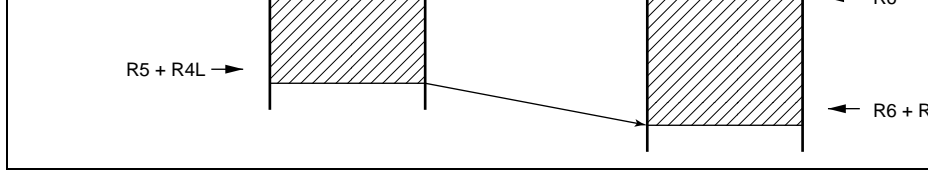
After executing BCLR:

	P47	P46	P45	P44	P43	P42	P41
Input/output	Output	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
DDR	1	1	1	1	1	1	1
DR	1	0	0	0	0	0	0

Operation:

1. When the BCLR instruction is executed, first the CPU reads P4DDR.
Since P4DDR is a write-only register, so the CPU reads H'FF. In this example P4DDR has a value of H'3F, but the value read by the CPU is H'FF.
2. The CPU clears bit 0 of the read data to 0, changing data to H'FE.
3. The CPU writes H'FE to DDR, completing execution of BCLR.

As a result of the BCLR instruction, bit 0 in DDR is set to 0, and P40 becomes an input pin. However, bits 7 and 6 of DDR are modified to 1, therefore P47 and P46 become output pins.



- Set $R4L$ and $R6$ so that the end address of the destination address (value of $R6 + R4L$) not exceed $H'FFFF$ (the value of $R6$ must not change from $H'FFFF$ to $H'0000$ during execution).

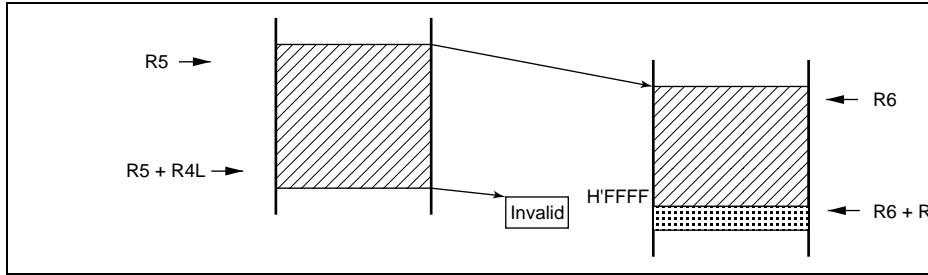


Table 3.1 MCU Operating Mode Selection

MCU Operating Mode	MD1	MD0	CPU Operating Mode	Description
0	0	0	—	—
1		1	Normal	Expanded mode with on-chip ROM disabled
2	1	0	Advanced	Expanded mode with on-chip ROM enabled Single-chip mode
3		1	Normal	Expanded mode with on-chip ROM enabled Single-chip mode

Mode 1 is an expanded mode that allows access to external memory and peripheral devices. In modes 2 and 3, operation begins in single-chip mode after reset release, but a transition is made to external expansion mode by setting the EXPE bit in MDCR to 1.

Mode 0 cannot be used in this LSI. Thus, mode pins should be set to enable mode 1, 2, or 3 as the normal program execution state. Mode pins should not be changed during operation.

3.2 Register Descriptions

The following registers are related to the operating mode. For details on the bus control registers (BCR), refer to section 6.3.1, Bus Control Register (BCR).

- Mode control register (MDCR)
- System control register (SYSCR)
- Serial timer control register (STCR)

0: Single-chip mode

1: Extended mode

6 to 2	—	All 0	R	Reserved
				These bits are always read as 0. These bits are not modified.
1	MDS1	—*	R	Mode Select 1 and 0
0	MDS0	—*	R	These bits indicate the input levels at mode select pins (MD1 and MD0) (the current operating mode). Bits MDS1 and MDS0 correspond to MD1 and MD0, respectively. These bits are read-only bits and they cannot be written to. The mode pin (MD1 and MD0) input levels are latched into these bits when MDCR is reset. These latches are canceled by a reset.

Note: * The initial values are determined by the settings of the MD1 and MD0 pins.

Specifies the location of the control pin (\overline{CS}) of the host interface together with the FGA20E bit. See section 18, Host Interface X-Bus Interface for details.

6	IOSE	0	R/W	<p>IOS Enable</p> <p>Enables or disables $\overline{AS}/\overline{IOS}$ pin function in interrupt mode.</p> <p>0: \overline{AS} pin Outputs low when an external area is accessed.</p> <p>1: \overline{IOS} pin Outputs low when a specified address of H'(FF)F000 to H'(FF)F7FF is accessed.</p>
5	INTM1	0	R	<p>These bits select the control mode of the interrupt controller. For details on the interrupt control modes and interrupt control select modes 1 and 0, see section 5.6, Interrupt Control Modes and Interrupt Control Select Modes.</p> <p>00: Interrupt control mode 0</p> <p>01: Interrupt control mode 1</p> <p>10: Setting prohibited</p> <p>11: Setting prohibited</p>
4	INTM0	0	R/W	
3	XRST	1	R	<p>External Reset</p> <p>This bit indicates the reset source. A reset is caused by an external reset input, or when the watchdog timer overflows.</p> <p>0: A reset is caused when the watchdog timer overflows.</p> <p>1: A reset is caused by an external reset.</p>

Controls CPU access to the host interface registers (HICR, IDR1, ODR1, STR1, IDR2, ODR2, and STR2), the keyboard matrix interrupt and MOS input pull-up control registers (KMIMR, KMPCR, and KMPDR), the 8-bit timer (TMR_X and TMR_Y) registers (TCR_X/TCR_Y, TCSR_X/TCSR_Y, TICRR/TCORA_Y, TICRF/TCORB_Y, TCNT_X/TCNT_Y, TCORC/TISR, TCORA_Y, TCORB_X), and the timer connection registers (TCONRI, TCONRO, TCONRS, and SEDG).

0: In areas H'(FF)FFF0 to H'(FF)FFF7 and H'(FF)FFF8 to H'(FF)FFFF, CPU access to 8-bit timer registers and timer connection registers is permitted

1: In areas H'(FF)FFF0 to H'(FF)FFF7 and H'(FF)FFF8 to H'(FF)FFFF, CPU access to host interface registers and keyboard matrix interrupt and input pull-up control registers is permitted

0	RAME	1	R/W	RAM Enable
---	------	---	-----	------------

Enables or disables on-chip RAM. The RAM is initialized when the reset state is released.

0: On-chip RAM is disabled
1: On-chip RAM is enabled

3.2.3 Serial Timer Control Register (STCR)

STCR enables or disables register access, IIC operating mode, and on-chip flash memory. STCR also selects the input clock of the timer counter.

6	IICX1	0	R/W	I ² C Transfer Rate Select 1 and 0
5	IICX0	0	R/W	These bits control the IIC operation. They control the transfer rate in master mode together with IICX1. For details on the transfer rate, refer to the IIC mode registers CKS2 to CKS0 in the I ² C bus mode register. For details on the transfer rate, refer to the IIC mode registers.
4	IICE	0	R/W	<p>I²C Master Enable</p> <p>Enables or disables CPU access for IIC registers (ICCR, ICSR, ICDR/SARX, ICMR/SAR), registers (DADRAH/DACR, DADRAL, DADRBH/DACNTH, DADRBL/DACNTL), registers (SMR, BRR, SCMR).</p> <p>0: SCI_1 registers are accessed in an address range from H'(FF)FF88 to H'(FF)FF89 and from H'(FF)FF80 to H'(FF)FF8F.</p> <p>SCI_2 registers are accessed in an address range from H'(FF)FFA0 to H'(FF)FFA1 and from H'(FF)FFA2 to H'(FF)FFA7.</p> <p>SCI_0 registers are accessed in an address range from H'(FF)FFD8 to H'(FF)FFD9 and from H'(FF)FFD0 to H'(FF)FFDF.</p> <p>1: IIC_1 registers are accessed in an address range from H'(FF)FF88 to H'(FF)FF89 and from H'(FF)FF80 to H'(FF)FF8F.</p> <p>PWMX registers are accessed in an address range from H'(FF)FFA0 to H'(FF)FFA1 and from H'(FF)FFA2 to H'(FF)FFA7.</p> <p>IIC_0 registers are accessed in an address range from H'(FF)FFD8 to H'(FF)FFD9 and from H'(FF)FFD0 to H'(FF)FFDF.</p>

accessed in an area from H'(FF)FF80 to H'(FF)FF87.

1: Control registers of flash memory are accessed in an area from H'(FF)FF80 to H'(FF)FF87.

2	—	0	R/(W)	Reserved The initial value should not be changed.
1	ICKS1	0	R/W	Internal Clock Source Select 1, 0
0	ICKS0	0	R/W	These bits select a clock to be input to the counter (TCNT) and a count condition toggle bits CKS2 to CKS0 in the timer control register (TCR). For details, refer to section 12.3.4, Control Register (TCR).

3.3.2 Mode 2

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM

After a reset, the LSI is set to single-chip mode. To access an external address space, the MDCR should be set to 1.

When the EXPE bit in MDCR is set to 1, ports 1, 2 and A function as input ports after a reset. Ports 1, 2 and A output an address by setting 1 to the corresponding port data direction register (DDR). Port 3 functions as a data bus, and parts of port 9 carry bus control signals. Port 9 functions as a data bus when the ABW bit in WSCR is cleared to 0.

3.3.3 Mode 3

The CPU can access a 64-kbyte address space in normal mode. The on-chip ROM is 16-kbytes. The CPU can access a 56-kbyte address space in mode 3.

After a reset, the LSI is set to single-chip mode. To access an external address space, the MDCR should be set to 1.

When the EXPE bit in MDCR is set to 1, ports 1 and 2 function as input ports after a reset. Ports 1 and 2 function as an address bus by setting 1 to the corresponding port data direction register (DDR). Port 3 functions as a data bus, and parts of port 9 carry bus control signals. Port 9 functions as a data bus when the ABW bit in WSCR is cleared to 0.

Port 2		A	P*/A	P*/A
Port A		P	P*/A	P
Port 3		D	P*/D	P*/D
Port B		P*/D	P*/D	P*/D
Port 9	P97	P*/C	P*/C	P*/C
	P96	C*/P	P*/C	P*/C
	P95 to P93	C	P*/C	P*/C
	P92, P91	P	P	P
	P90	P*/C	P*/C	P*/C
Port C to Port G		P	P	P

Legend:

P: I/O port

A: Address bus output

D: Data bus I/O

C: Control signals, clock I/O

*: Immediately after reset

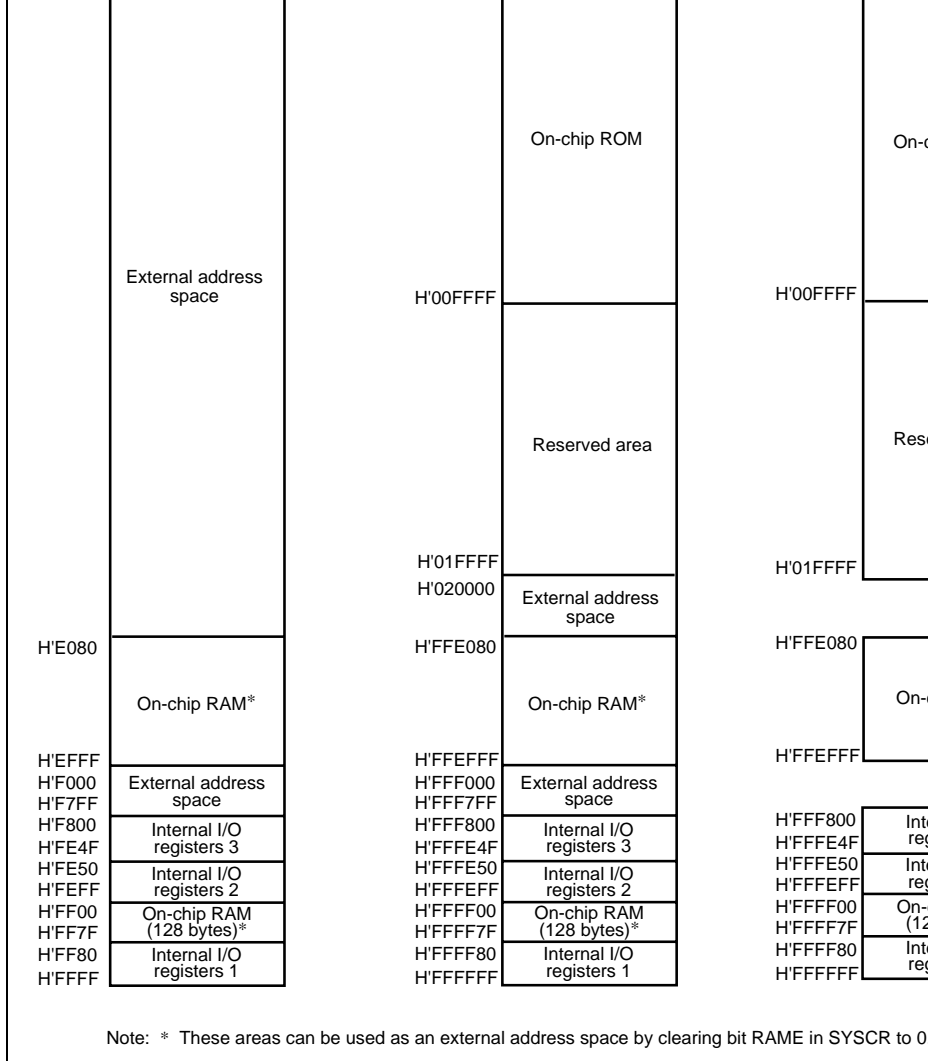


Figure 3.1 Address Map for H8S/2140B and H8S/2160B (1)

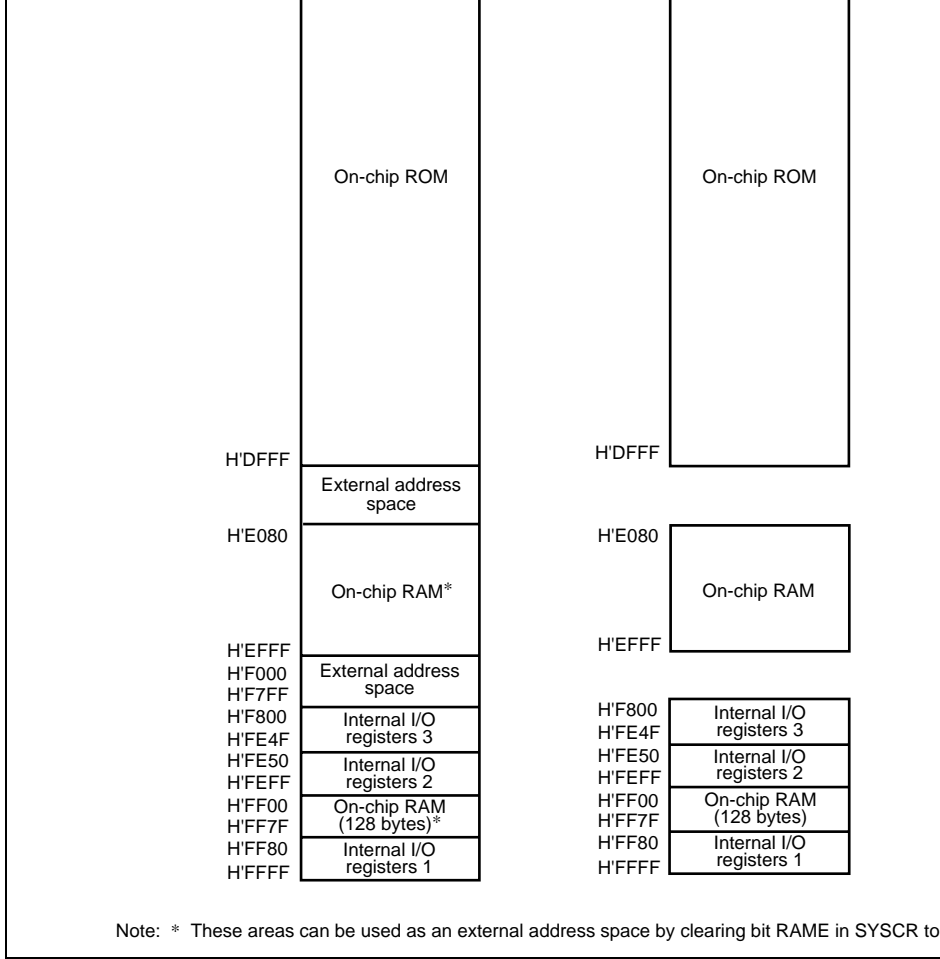


Figure 3.2 Address Map for H8S/2140B and H8S/2160B (2)

	External address space		On-chip ROM		On-
				H'01FFFF	H'01FFFF
				H'020000	
H'E080			External address space	H'FFE080	H'FFE080
	On-chip RAM*		On-chip RAM*		On-
H'EFFF		H'FEFFF		H'FEFFF	H'FEFFF
H'F000	External address space	H'FF000	External address space		
H'F7FF		H'FF7FF			
H'F800	Internal I/O registers 3	H'FF800	Internal I/O registers 3	H'FF800	Int reg
H'FE4F		H'FFE4F		H'FFE4F	H'FFE4F
H'FE50	Internal I/O registers 2	H'FE50	Internal I/O registers 2	H'FE50	Int reg
H'FEFF		H'FEFF		H'FEFF	H'FEFF
H'FF00	On-chip RAM (128 bytes)*	H'FF00	On-chip RAM (128 bytes)*	H'FF00	On- (12
H'FF7F		H'FF7F		H'FF7F	H'FF7F
H'FF80	Internal I/O registers 1	H'FF80	Internal I/O registers 1	H'FF80	Int reg
H'FFFF		H'FFFF		H'FFFF	H'FFFF

Note: * These areas can be used as an external address space by clearing bit RAME in SYSCR to 0

Figure 3.3 Address Map for H8S/2141B and H8S/2161B (1)

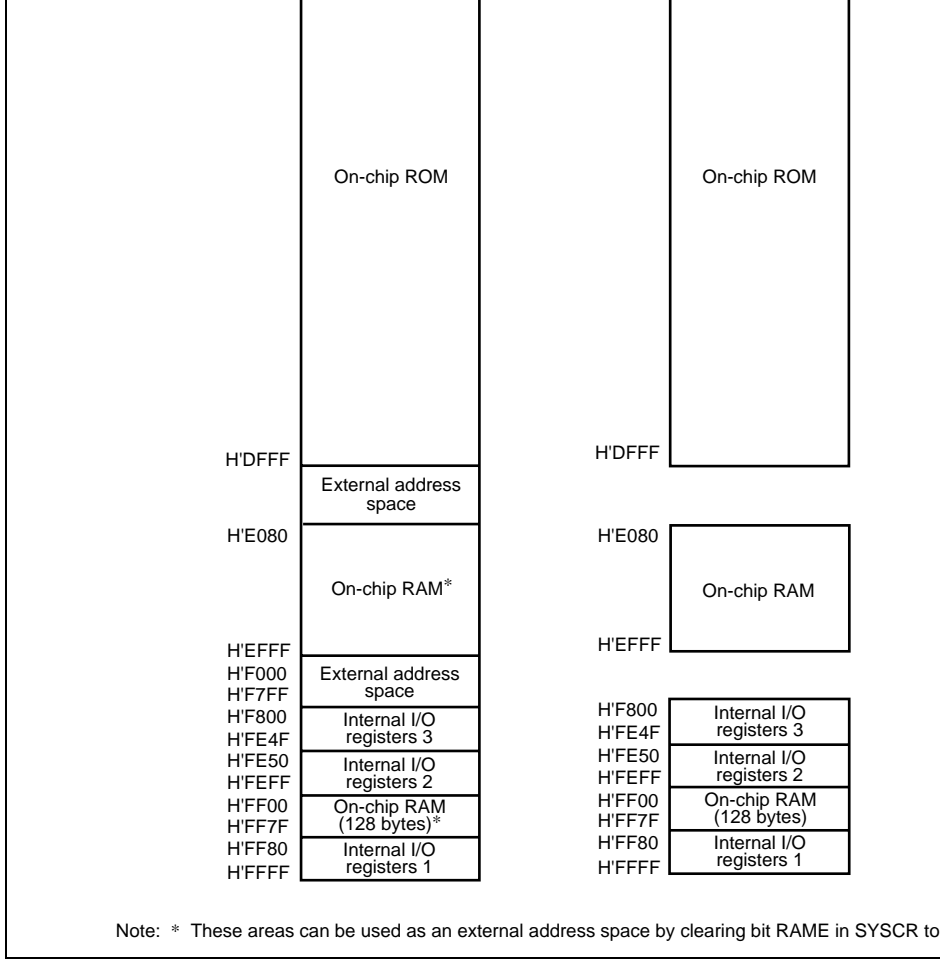


Figure 3.4 Address Map for H8S/2141B and H8S/2161B (2)

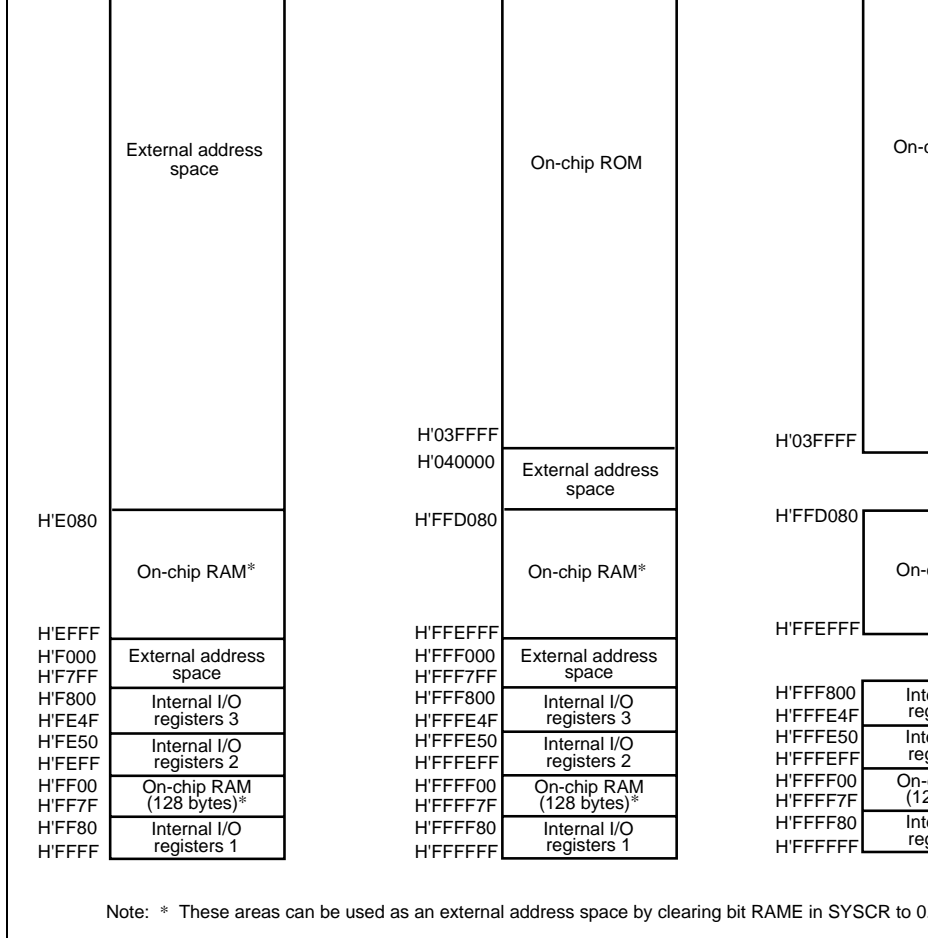


Figure 3.5 Address Map for H8S/2145BV (1)

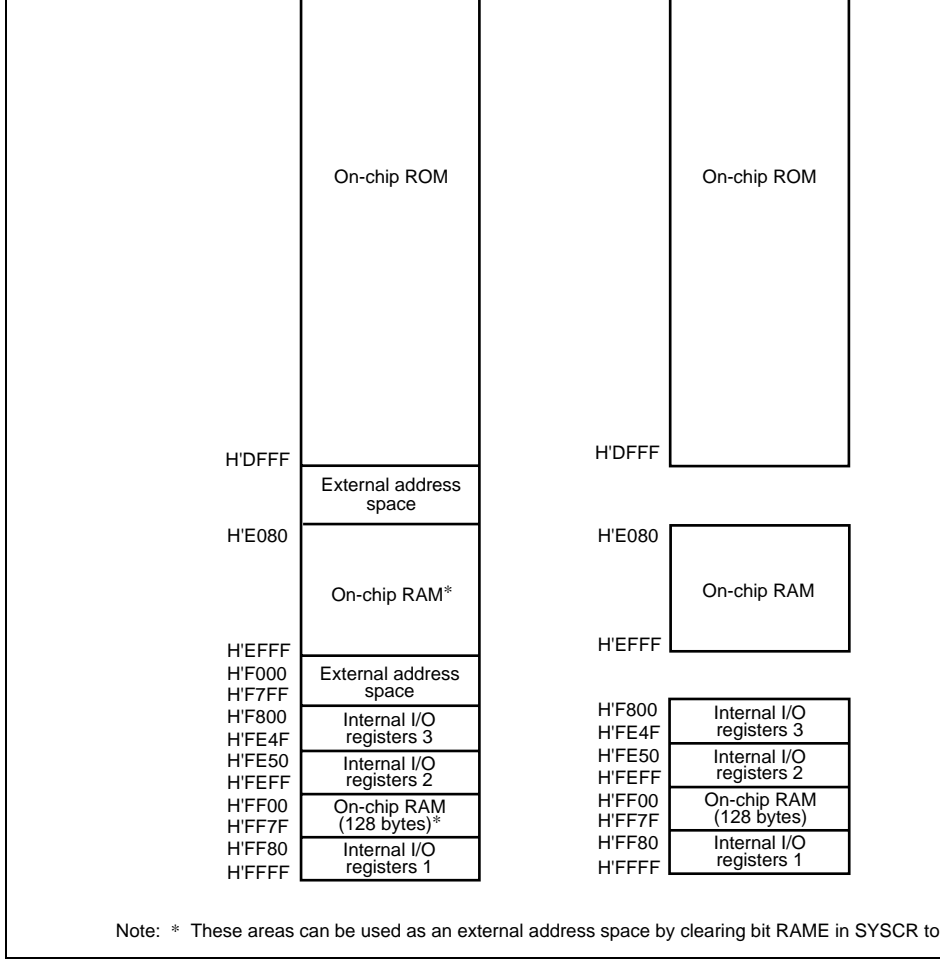


Figure 3.6 Address Map for H8S/2145BV (2)

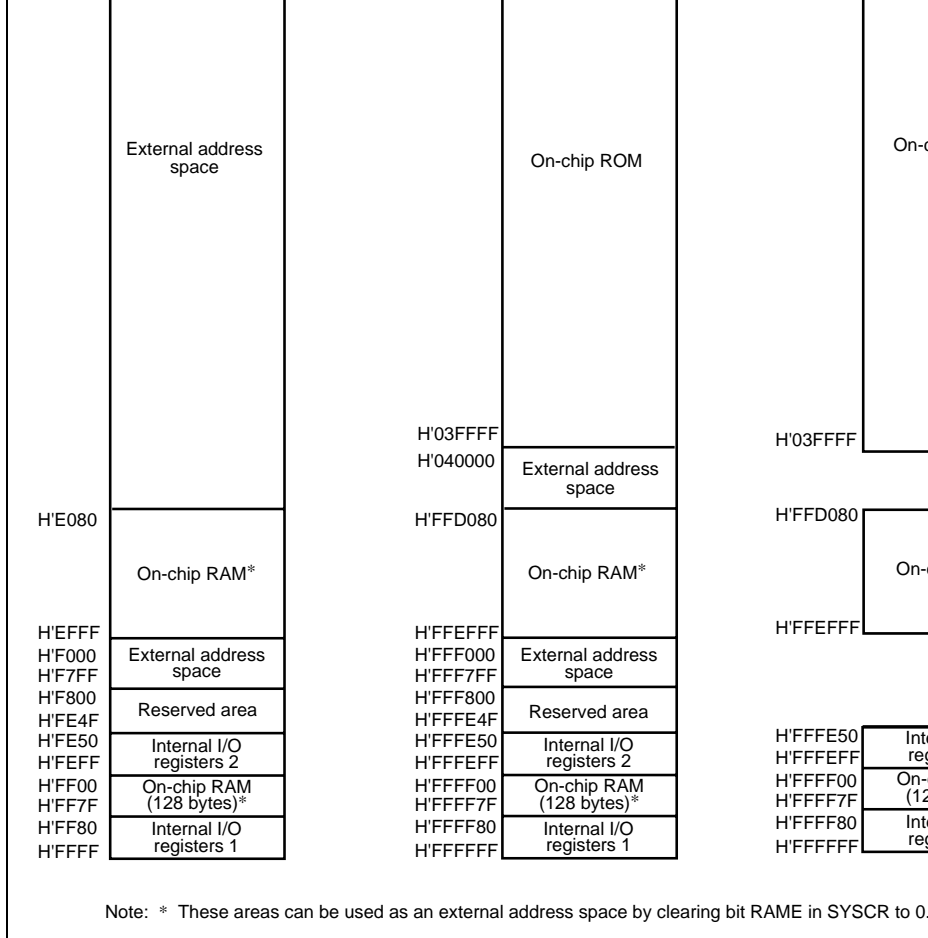


Figure 3.7 Address Map for H8S/2145B (1)

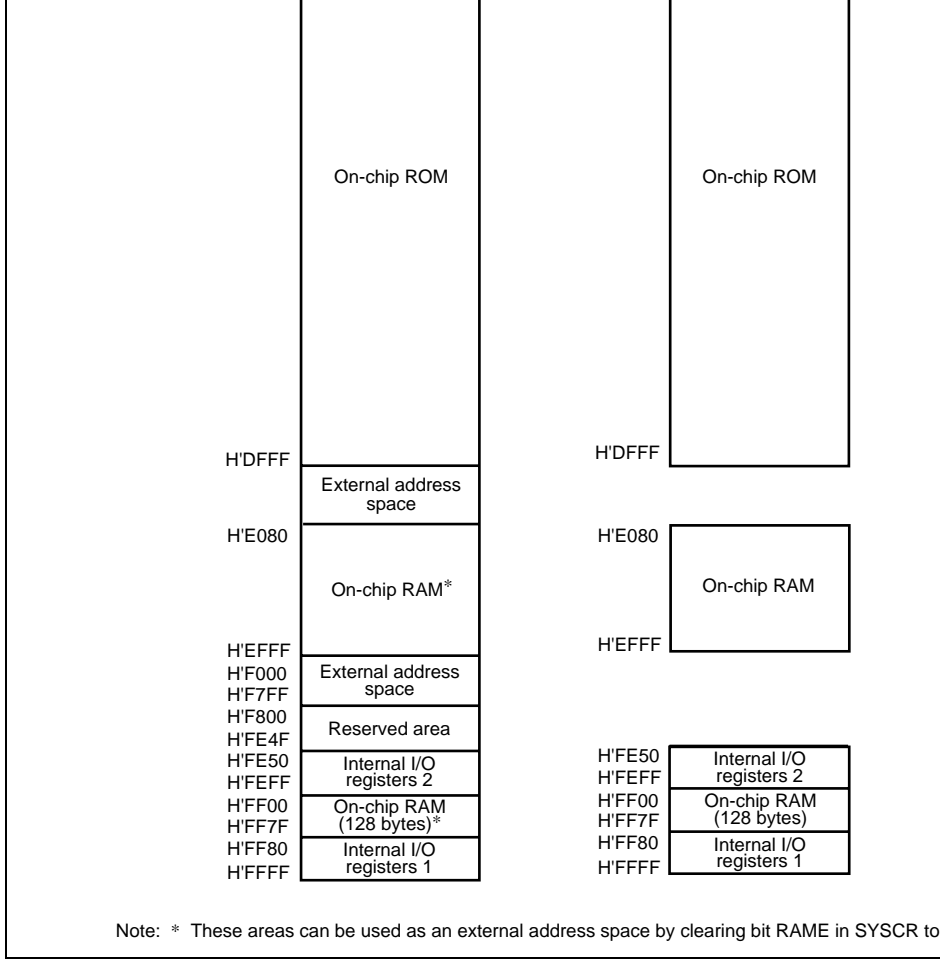


Figure 3.8 Address Map for H8S/2145B (2)

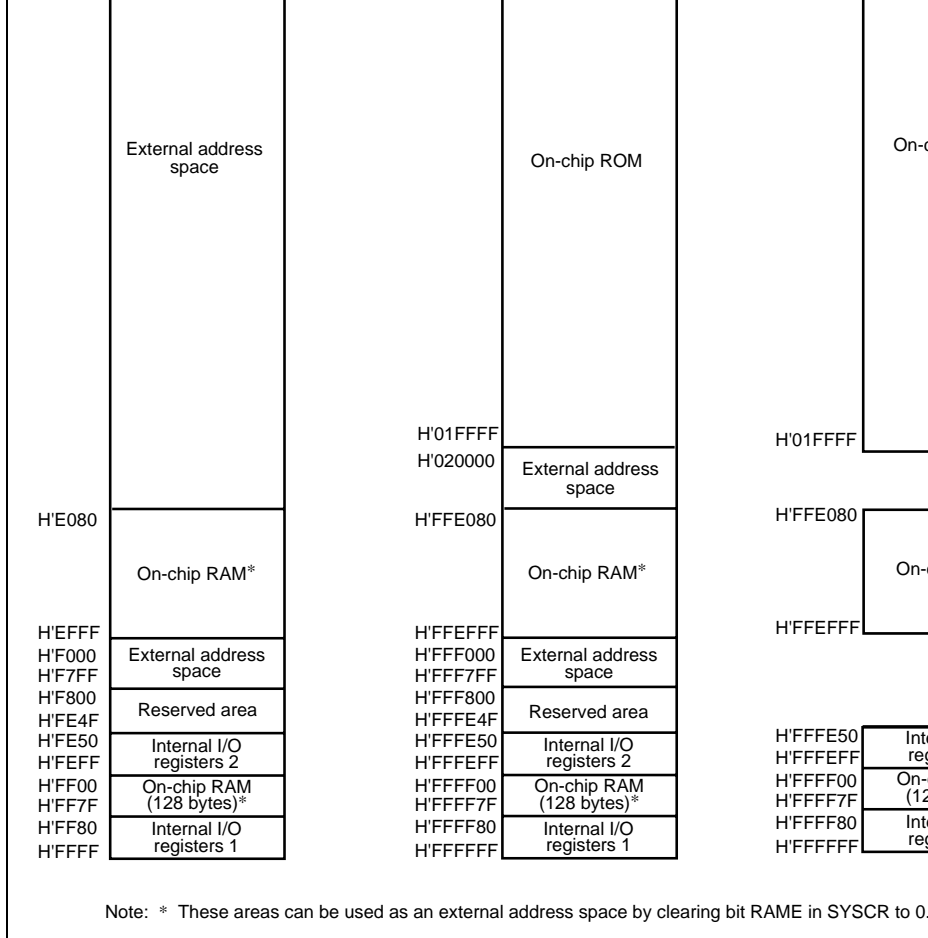


Figure 3.9 Address Map for H8S/2148B (1)

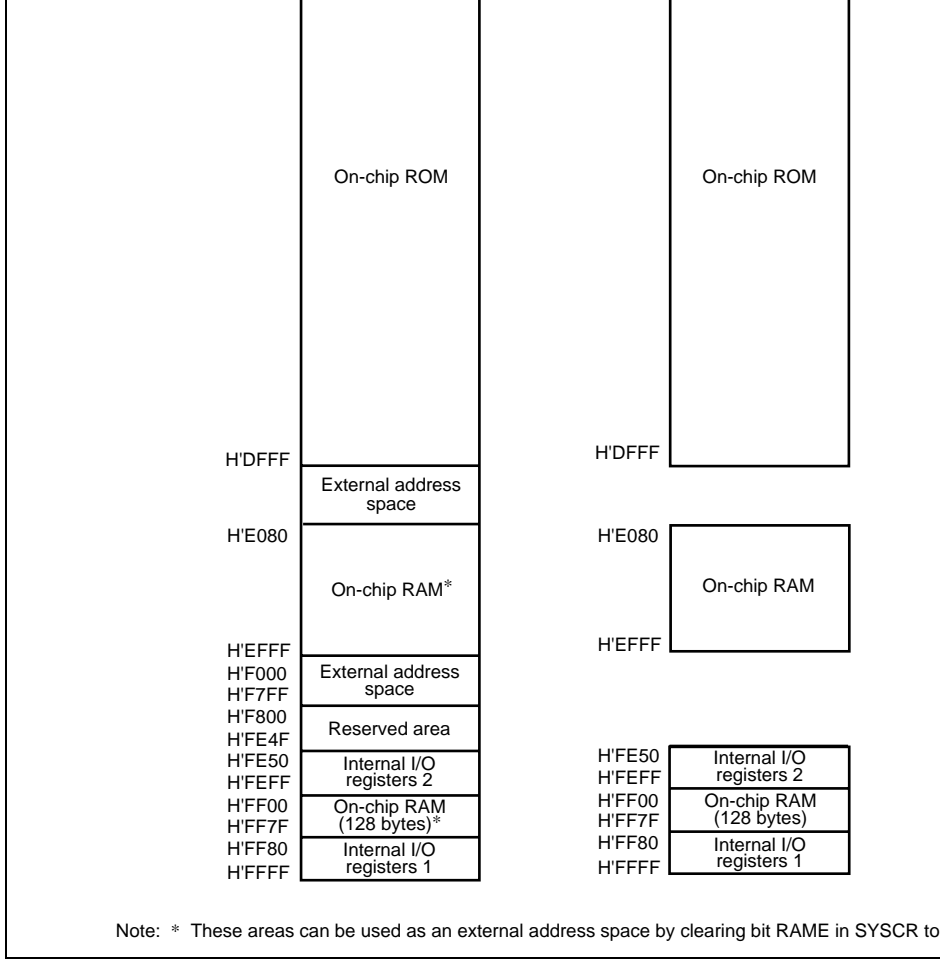


Figure 3.10 Address Map for H8S/2148B (2)

Table 4.1 Exception Types and Priority

Priority	Exception Type	Start of Exception Handling
High ↑ Low	Reset	Starts immediately after a low-to-high transition pin, or when the watchdog timer overflows.
	Interrupt	Starts when execution of the current instruction handling ends, if an interrupt request has been detected. Interrupt detection is not performed on completion of ORC, XORC, or LDC instruction execution, or on completion of reset exception handling.
	Direct transition	Starts when a direction transition occurs as the SLEEP instruction execution.
	Trap instruction	Started by execution of a trap (TRAPA) instruction. Instruction exception handling requests are accepted at any times in program execution state.

Exception Source	Vector Number	Normal Mode	Advanced Mode	
Reset	0	H'0000 to H'0001	H'000000 to H'000001	
Reserved for system use	1	H'0002 to H'0003	H'000004 to H'000005	
	5	H'000A to H'000B	H'000014 to H'000015	
Direct transition	6	H'000C to H'000D	H'000018 to H'000019	
External interrupt (NMI)	7	H'000E to H'000F	H'00001C to H'00001D	
Trap instruction (four sources)	8	H'0010 to H'0011	H'000020 to H'000021	
	9	H'0012 to H'0013	H'000024 to H'000025	
	10	H'0014 to H'0015	H'000028 to H'000029	
	11	H'0016 to H'0017	H'00002C to H'00002D	
Reserved for system use	12	H'0018 to H'0019	H'000030 to H'000031	
	15	H'001E to H'001F	H'00003C to H'00003D	
External interrupt	IRQ0	16	H'0020 to H'0021	H'000040 to H'000041
	IRQ1	17	H'0022 to H'0023	H'000044 to H'000045
	IRQ2	18	H'0024 to H'0025	H'000048 to H'000049
	IRQ3	19	H'0026 to H'0027	H'00004C to H'00004D
	IRQ4	20	H'0028 to H'0029	H'000050 to H'000051
	IRQ5	21	H'002A to H'002B	H'000054 to H'000055
	IRQ6	22	H'002C to H'002D	H'000058 to H'000059
	IRQ7	23	H'002E to H'002F	H'00005C to H'00005D
Internal interrupt*	24	H'0030 to H'0031	H'000060 to H'000061	
	107	H'00DE to H'00DF	H'0001BC to H'0001BD	

Note: * For details on the internal interrupt vector table, see section 5.5, Interrupt Exception Handling Vector Table.

4.3.1 Reset Exception Handling

When the $\overline{\text{RES}}$ pin goes high after being held low for the necessary time, this LSI starts exception handling as follows:

1. The internal state of the CPU and the registers of the on-chip peripheral modules are reset, and the I bit is set to 1 in CCR.
2. The reset exception handling vector address is read and transferred to the PC, and execution starts from the address indicated by the PC.

Figure 4.1 shows an example of the reset sequence.

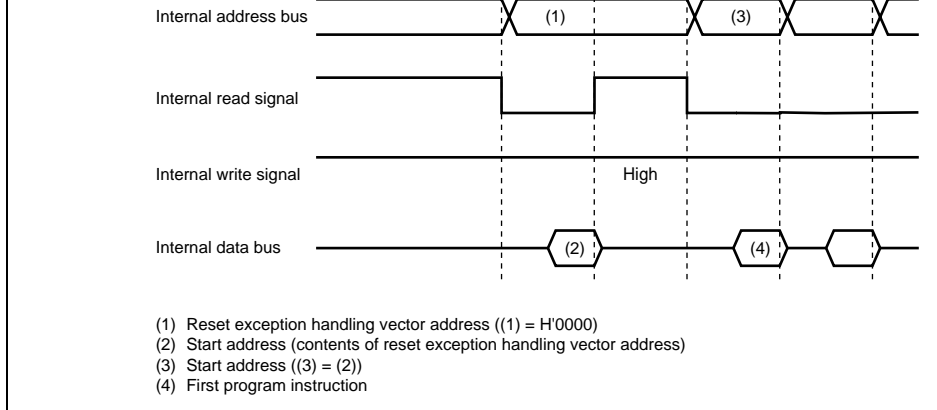


Figure 4.1 Reset Sequence (Mode 3)

4.3.2 Interrupts after Reset

If an interrupt is accepted after a reset and before the stack pointer (SP) is initialized, the CCR will not be saved correctly, leading to a program crash. To prevent this, all interrupts including NMI, are disabled immediately after a reset. Since the first instruction of a program is always executed immediately after the reset state ends, make sure that this instruction initializes the stack pointer (example: `MOV.L #xx: 32, SP`).

4.3.3 On-Chip Peripheral Modules after Reset Is Cancelled

After a reset is cancelled, the module stop control registers (MSTPCR) are initialized, and all peripheral modules except the DTC operate in module stop mode. Therefore, the registers of on-chip peripheral modules cannot be read from or written to. To read from and write to these registers, clear module stop mode.

1. The values in the program counter (PC) and condition code register (CCR) are saved on the stack.
2. A vector address corresponding to the interrupt source is generated, the start address is loaded from the vector table to the PC, and program execution begins from that address.

4.5 Trap Instruction Exception Handling

Trap instruction exception handling starts when a TRAPA instruction is executed. Trap instruction exception handling can be executed at all times in the program execution state.

Trap instruction exception handling is conducted as follows:

1. The values in the program counter (PC) and condition code register (CCR) are saved on the stack.
2. A vector address corresponding to the interrupt source is generated, the start address is loaded from the vector table to the PC, and program execution starts from that address.

The TRAPA instruction fetches a start address from a vector table entry corresponding to the interrupt number from 0 to 3, as specified in the instruction code.

Table 4.3 shows the status of CCR after execution of trap instruction exception handling.

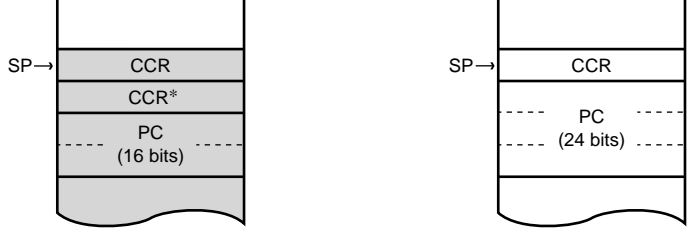
Table 4.3 Status of CCR after Trap Instruction Exception Handling

Interrupt Control Mode	CCR	
	I	UI
0	1	—
1	1	1

Legend:

1: Set to 1

—: Retains value prior to execution



Note: * Ignored on return.

Figure 4.2 Stack Status after Exception Handling

PUSH.W Rn (or MOV.W Rn, @-SP)
 PUSH.L ERn (or MOV.L ERn, @-SP)

Use the following instructions to restore registers:

POP.W Rn (or MOV.W @SP+, Rn)
 POP.L ERn (or MOV.L @SP+, ERn)

Setting SP to an odd value may lead to a malfunction. Figure 4.3 shows an example of what happens when the SP value is odd.

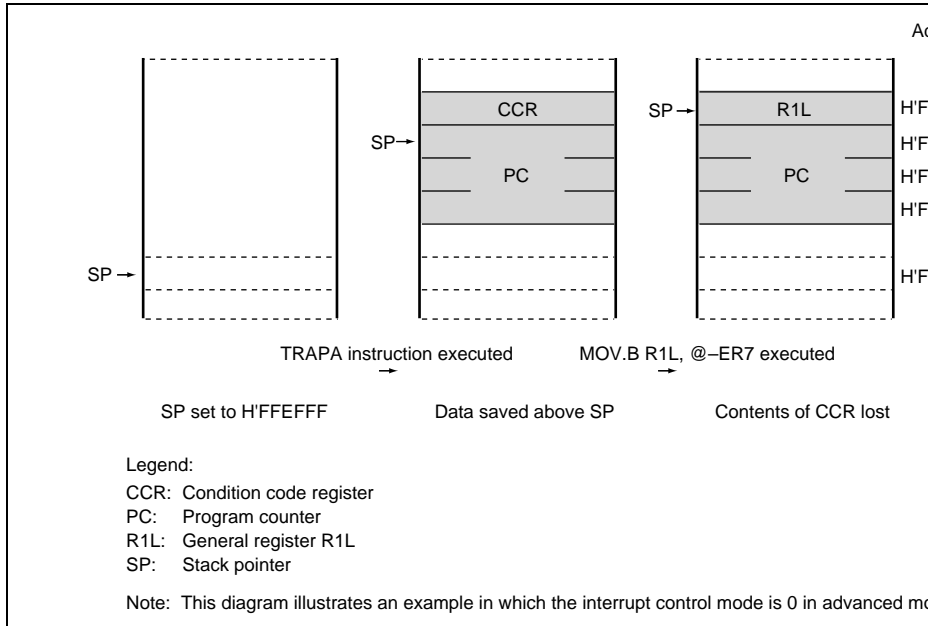


Figure 4.3 Operation when SP Value Is Odd

- Priorities settable with ICR
An interrupt control register (ICR) is provided for setting interrupt priorities. Three levels can be set for each module for all interrupts except NMI and address break.
- Independent vector addresses
All interrupt sources are assigned independent vector addresses, making it unnecessary for a source to be identified in the interrupt handling routine.
- Thirty-one external interrupts
NMI is the highest-priority interrupt, and is accepted at all times. Rising edge or falling-edge detection can be selected for NMI. Falling-edge, rising-edge, or both-edge detection sensing, can be selected for $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$. The $\overline{\text{IRQ6}}$ interrupt is shared by the interrupt from the $\overline{\text{IRQ6}}$ pin and eight external interrupt inputs ($\overline{\text{KIN7}}$ to $\overline{\text{KIN0}}$), and the $\overline{\text{IRQ7}}$ interrupt is shared by the interrupt from the $\overline{\text{IRQ7}}$ pin and sixteen external interrupt inputs ($\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$ and $\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$). $\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$ and $\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$ can be masked individually by the user program.
- DTC control
The DTC can be activated by an interrupt request.

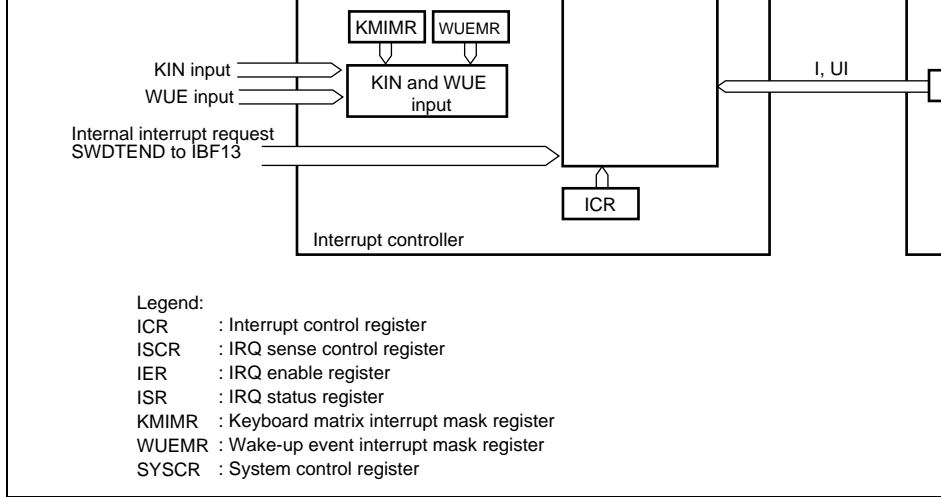


Figure 5.1 Block Diagram of Interrupt Controller

$\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$	Input	Maskable external interrupts Rising edge, falling edge, or both edges, or level sensing can be selected individually for each pin.
$\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$	Input	Maskable external interrupts Falling edge or level sensing can be selected.
$\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$ *	Input	Maskable external interrupts Falling edge or level sensing can be selected.

Note: * Not supported by the H8S/2148B and H8S/2145B (5-V version).

5.3 Register Descriptions

The interrupt controller has the following registers. For details on the system control register (SYSCR), refer to section 3.2.2, System Control Register (SYSCR).

- Interrupt control registers A to C (ICRA to ICRC)
- Address break control register (ABRKCR)
- Break address registers A to C (BARA to BARC)
- IRQ sense control registers (ISCRH, ISCRL)
- IRQ enable register (IER)
- IRQ status register (ISR)
- Keyboard matrix interrupt mask registers (KMIMRA, KMIMR)
- Wake-up event interrupt mask register (WUEMRB)

to to
0 ICRn0

0: Corresponding interrupt source is
control level 0 (no priority)
1: Corresponding interrupt source is
control level 1 (priority)

n: A to C

Table 5.2 Correspondence between Interrupt Source and ICR

Bit	Bit Name	Register		
		ICRA	ICRB	ICRC
7	ICRn7	IRQ0	A/D converter	SCI_0
6	ICRn6	IRQ1	FRT	SCI_1
5	ICRn5	IRQ2, IRQ3	—	SCI_2
4	ICRn4	IRQ4, IRQ5	—	IIC_0
3	ICRn3	IRQ6, IRQ7	TMR_0	IIC_1
2	ICRn2	DTC	TMR_1	—
1	ICRn1	WDT_0	TMR_X, TMR_Y	LPC*
0	ICRn0	WDT_1	XBS, Keyboard buffer controller	—

Legend:

—: Reserved. The write value should always be 0.

Notes: n: A to C

* On products not including LPC, this bit is reserved. The write value should a

address specified by BARA to BAR
prefetched.

[Setting condition]

When an address specified by BAR
is prefetched while the BIE flag is s

[Clearing condition]

When an exception handling is exe
address break interrupt.

6 to 1	—	All 0	R	Reserved These bits are always read as 0 and modified.
0	BIE	0	R/W	Break Interrupt Enable Enables or disables address break. 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	A23 to A16	All 0	R/W	Addresses 23 to 16 The A23 to A16 bits are compared with A16 in the internal address bus.

- BARB

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	A15 to A8	All 0	R/W	Addresses 15 to 8 The A15 to A8 bits are compared with A8 in the internal address bus.

- BARC

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	A7 to A1	All 0	R/W	Addresses 7 to 1 The A7 to A1 bits are compared with in the internal address bus.
0	—	0	R	Reserved This bit is always read as 0 and cannot be modified.

6	IRQ7SCA	0	R/W	IRQn Sense Control A
5	IRQ6SCB	0	R/W	00: Interrupt request generated at $\overline{\text{IRQn}}$ input
4	IRQ6SCA	0	R/W	
3	IRQ5SCB	0	R/W	01: Interrupt request generated at $\overline{\text{IRQn}}$ input
2	IRQ5SCA	0	R/W	
1	IRQ4SCB	0	R/W	10: Interrupt request generated at $\overline{\text{IRQn}}$ input
0	IRQ4SCA	0	R/W	11: Interrupt request generated at $\overline{\text{IRQn}}$ and rising edges of $\overline{\text{IRQn}}$ input (n = 7 to 4)

- ISCR_L

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ3SCB	0	R/W	IRQn Sense Control B
6	IRQ3SCA	0	R/W	IRQn Sense Control A
5	IRQ2SCB	0	R/W	00: Interrupt request generated at $\overline{\text{IRQn}}$ input
4	IRQ2SCA	0	R/W	
3	IRQ1SCB	0	R/W	01: Interrupt request generated at $\overline{\text{IRQn}}$ input
2	IRQ1SCA	0	R/W	
1	IRQ0SCB	0	R/W	10: Interrupt request generated at $\overline{\text{IRQn}}$ input
0	IRQ0SCA	0	R/W	11: Interrupt request generated at $\overline{\text{IRQn}}$ and rising edges of $\overline{\text{IRQn}}$ input (n = 3 to 0)

4	IRQ4E	0	R/W
3	IRQ3E	0	R/W
2	IRQ2E	0	R/W
1	IRQ1E	0	R/W
0	IRQ0E	0	R/W

5.3.6 IRQ Status Register (ISR)

The ISR register is a flag register that indicates the status of IRQ7 to IRQ0 interrupt re

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ7F	0	R/(W) ^{*2}	[Setting condition]
6	IRQ6F	0	R/(W) ^{*2}	When the interrupt source selected b
5	IRQ5F	0	R/(W) ^{*2}	registers occurs
4	IRQ4F	0	R/(W) ^{*2}	[Clearing conditions]
3	IRQ3F	0	R/(W) ^{*2}	<ul style="list-style-type: none"> When reading IRQnF flag when then writing 0 to IRQnF flag
2	IRQ2F	0	R/(W) ^{*2}	<ul style="list-style-type: none"> When interrupt exception handlin executed when low-level detectio
1	IRQ1F	0	R/(W) ^{*2}	<ul style="list-style-type: none"> and $\overline{\text{IRQn}}$ input is high (n = 7 to
0	IRQ0F	0	R/(W) ^{*2}	<ul style="list-style-type: none"> When IRQn interrupt exception h executed when falling-edge, risin both-edge detection is set^{*1}

Notes: 1. When a product, in which a DTC is incorporated, is used, the corresponding not automatically cleared even when exception handing is executed. For det section 5.8.4, Setting on a Product Incorporating DTC.

2. Only 0 can be written, for flag clearing.

7	KMIMR15	1	R/W	Keyboard Matrix Interrupt Mask 15
6	KMIMR14	1	R/W	These bits enable or disable a key-input interrupt request (KIN15 to KIN14)
5	KMIMR13	1	R/W	
4	KMIMR12	1	R/W	0: Enables a key-sensing input interrupt request
3	KMIMR11	1	R/W	1: Disables a key-sensing input interrupt request
2	KMIMR10	1	R/W	
1	KMIMR9	1	R/W	
0	KMIMR8	1	R/W	

- KMIMR

Bit	Bit Name	Initial Value	R/W	Description
7	KMIMR7	1	R/W	Keyboard Matrix Interrupt Mask 7 to 0
6	KMIMR6	0	R/W	These bits enable or disable a key-input interrupt request (KIN7 to KIN6). KMIMR6 also performs interrupt request control for pin IRQ6.
5	KMIMR5	1	R/W	
4	KMIMR4	1	R/W	0: Enables a key-sensing input interrupt request
3	KMIMR3	1	R/W	1: Disables a key-sensing input interrupt request
2	KMIMR2	1	R/W	
1	KMIMR1	1	R/W	
0	KMIMR0	1	R/W	

2	WUEMR2	1	R/W	1: Disables a wake-up event input request
1	WUEMR1	1	R/W	
0	WUEMR0	1	R/W	

Note: * Not supported by the H8S/2148B and H8S/2145B (5-V version).

Figure 5.2 shows the relationship between interrupts IRQ7 and IRQ6, interrupts KIN15 interrupts WUE7 to WUE0, and registers KMIMRA, KMIMR, and WUEMRB.

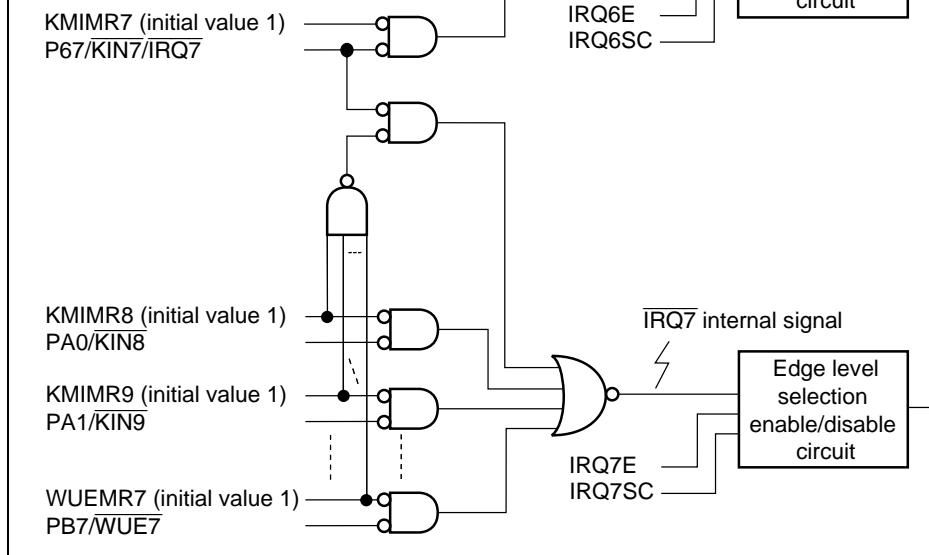


Figure 5.2 Relationship between Interrupts IRQ7 and IRQ6, Interrupts KIN15 to KIN0, Interrupts WUE7 to WUE0, and Registers KMIMR, KMIMRA, and WUEMR

If any of bits KMIMR15 to KMIMR8 or WUEMRB7 to WUEMRB0 is cleared to 0, the $\overline{\text{IRQ7}}$ input from the $\overline{\text{IRQ7}}$ pin will be ignored. When pins KIN7 to KIN0 , KIN15 to KIN8 , WUE7 to WUE0 are used as key-sense interrupt input pins or wakeup event interrupt input pins, level sensing or falling-edge sensing must be designated as the interrupt sense condition for the corresponding interrupt source (IRQ6 or IRQ7).

NMI Interrupt: NMI is the highest-priority interrupt, and is always accepted by the CPU regardless of the interrupt control mode or the status of the CPU interrupt mask bits. The NMIEN bit in SYSCR can be used to select whether an interrupt is requested at a rising edge or a falling edge on the NMI pin.

IRQ7 to IRQ0 Interrupts: Interrupts IRQ7 to IRQ0 are requested by an input signal at pins $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ7}}$. Interrupts IRQ7 to IRQ0 have the following features:

- The interrupt exception handling for interrupt requests IRQ7 to IRQ0 can be started by a unique independent vector address.
- Using ISCR, it is possible to select whether an interrupt is generated by a low level, a falling edge, rising edge, or both edges, at pins $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$.
- Enabling or disabling of interrupt requests IRQ7 to IRQ0 can be selected with IER.
- Interrupt control levels can be specified by the ICR settings.
- The status of interrupt requests IRQ7 to IRQ0 is indicated in ISR. ISR flags can be cleared by software.

The detection of IRQ7 to IRQ0 interrupts does not depend on whether the relevant pin is set for input or output. However, when a pin is used as an external interrupt input pin, the corresponding DDR to 0 to use the pin as an I/O pin for another function.

A block diagram of interrupts IRQ7 to IRQ0 is shown in figure 5.3.

Figure 5.3 Block Diagram of Interrupts IRQ7 to IRQ0

When pin $\overline{\text{IRQ6}}$ is used as an IRQ6 interrupt input pin, clear the KMIMR6 bit to 0.

When pin $\overline{\text{IRQ7}}$ is used as an IRQ7 interrupt pin, set all of bits KMIMR15 to KMIMR7, WUEMR7 to WUEMR0 to 1. If any of these bits is cleared to 0, IRQ7 interrupt input $\overline{\text{IRQ7}}$ pin will be ignored.

Since interrupt request flags IRQ7F to IRQ0F are set each time the setting condition is met, regardless of the IER setting, refer to a needed flag only.

KIN15 to KIN0 Interrupts, WUE7 to WUE0 Interrupts: Interrupts KIN15 to KIN0 and WUE7 to WUE0 are requested by an input signal at pins $\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$ and $\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$. Pins $\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$ and $\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$ are used for key-sense input or wakeup event interrupt input pins. Corresponding KMIMR and WUEMR bits to 0 in order to enable their key-sense input or wakeup event interrupts. Remaining unused KMIMR and WUEMR bits for key-sense input or wakeup event interrupts should be set to 1 in order to disable interrupts. Interrupts WUE7 to WUE0 and KIN15 to KIN0 generate IRQ7 interrupts, and interrupts KIN7 to KIN0 generate IRQ6 interrupts. The conditions for interrupt request generation, enable of interrupt requests, settings of interrupt control levels, and status display of interrupt requests depend on each setting and display of IRQ7 or IRQ6 interrupt.

When pins $\overline{\text{KIN7}}$ to $\overline{\text{KIN0}}$, $\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$, or $\overline{\text{WUE7}}$ to $\overline{\text{WUE0}}$ are used as key-sense input pins or wakeup event interrupt input pins, either low-level sensing or falling-edge sensing must be designated as the interrupt sense condition for the corresponding interrupt source (IRQ6 or IRQ7).

2. The control level for each interrupt can be set by ICR.
3. The DTC can be activated by an interrupt request from an on-chip peripheral module.
4. An interrupt request that activates the DTC is not affected by the interrupt control mask bits or the status of the CPU interrupt mask bits.

5.5 Interrupt Exception Handling Vector Table

Table 5.3 lists interrupt exception handling sources, vector addresses, and interrupt priorities. For modules with default priorities, the lower the vector number, the higher the priority. Modules set at a higher priority will conform to their default priorities. Priorities within a module are fixed.

An interrupt control level can be specified for a module to which an ICR bit is assigned. Interrupt requests from modules that are set to control level 1 (priority) by the ICR bit setting and the UI bits in CCR are given priority and processed before interrupt requests from modules set to control level 0 (no priority).

	IRQ2	18	H'0024	H'000048	ICRA
	IRQ3	19	H'0026	H'00004C	
	IRQ4	20	H'0028	H'000050	ICRA
	IRQ5	21	H'002A	H'000054	
	IRQ6, KIN7 to KIN0	22	H'002C	H'000058	ICRA
	IRQ7, KIN15 to KIN8, WUE7 to WUE0	23	H'002E	H'00005C	
DTC	SWDTEND (Software activation data transfer end)	24	H'0030	H'000060	ICRA
WDT_0	WOVI0 (Interval timer)	25	H'0032	H'000064	ICRA
WDT_1	WOVI1 (Interval timer)	26	H'0034	H'000068	ICRA
—	Address break	27	H'0036	H'00006C	—
A/D converter	ADI (A/D conversion end)	28	H'0038	H'000070	ICRE
—	Reserved for system use	29 to 47	H'003A to H'005E	H'000074 to H'0000BC	—
FRT	ICIA (Input capture A)	48	H'0060	H'0000C0	ICRE
	ICIB (Input capture B)	49	H'0062	H'0000C4	
	ICIC (Input capture C)	50	H'0064	H'0000C8	
	ICID (Input capture D)	51	H'0066	H'0000CC	
	OCIA (Output compare A)	52	H'0068	H'0000D0	
	OCIB (Output compare B)	53	H'006A	H'0000D4	
	FOVI (Overflow)	54	H'006C	H'0000D8	
	Reserved for system use	55	H'006E	H'0000DC	
—	Reserved for system use	56 to 63	H'0070 to H'007E	H'0000E0 to H'0000FC	—
TMR_0	CMIA0 (Compare match A)	64	H'0080	H'000100	ICRE
	CMIB0 (Compare match A)	65	H'0082	H'000104	
	OVI0 (Overflow)	66	H'0084	H'000108	
	Reserved for system use	67	H'0086	H'00010C	

	OV11 (Overflow)	74	H'0094	H'000128	
	ICIX (Input capture X)	75	H'0096	H'00012C	
XBS	IBF1 (IDR1 reception completion)	76	H'0098	H'000130	ICRB
	IBF2 (IDR2 reception completion)	77	H'009A	H'000134	
	IBF3 (IDR3 reception completion)	78	H'009C	H'000138	
	IBF4 (IDR4 reception completion)	79	H'009E	H'00013C	
SCI_0	ERI0 (Reception error 0)	80	H'00A0	H'000140	ICRC
	RX10 (Reception completion 0)	81	H'00A2	H'000144	
	TX10 (Transmission data empty 0)	82	H'00A4	H'000148	
	TE10 (Transmission end 0)	83	H'00A6	H'00014C	
SCI_1	ERI1 (Reception error 1)	84	H'00A8	H'000150	ICRC
	RX11 (Reception completion 1)	85	H'00AA	H'000154	
	TX11 (Transmission data empty 1)	86	H'00AC	H'000158	
	TE11 (Transmission end 1)	87	H'00AE	H'00015C	
SCI_2	ERI2 (Reception error 2)	88	H'00B0	H'000160	ICRC
	RX12 (Reception completion 2)	89	H'00B2	H'000164	
	TX12 (Transmission data empty 2)	90	H'00B4	H'000168	
	TE12 (Transmission end 2)	91	H'00B6	H'00016C	
IIC_0	IIC10 (1-byte transmission/ reception completion)	92	H'00B8	H'000170	ICRC
	DDCSWI (Format switch)	93	H'00BA	H'000174	
IIC_1	IIC11 (1-byte transmission/ reception completion)	94	H'00BC	H'000178	ICRC
	Reserved for system use	95	H'00BE	H'00017C	
Keyboard buffer controller	KBIA (Reception completion A)	96	H'00C0	H'000180	ICRB
	KBIB (Reception completion B)	97	H'00C2	H'000184	
	KBIC (Reception completion C)	98	H'00C4	H'000188	
	Reserved for system use	99	H'00C6	H'00018C	
—	Reserved for system use	100	H'00C8	H'000190	—
		to	to	to	
		107	H'00D6	H'0001AC	
LPC*	ERR1 (Transfer error)	108	H'00D8	H'0001B0	ICRC
	IBF1 (IDR1 reception completion)	109	H'00DA	H'0001B4	
	IBF2 (IDR2 reception completion)	110	H'00DC	H'0001B8	
	IBF3 (IDR3 reception completion)	111	H'00DE	H'0001BC	

Note: * Reserved for system use on products not including LPC.

Interrupt Control Mode	SYSCR		Priority Setting Registers	Interrupt Mask Bits	Description
	INTM1	INTM0			
0	0	0	ICR	I	Interrupt mask control is performed by the I bit. Priority levels can be set with ICR.
1		1	ICR	I, UI	3-level interrupt mask control is performed by the I and UI bits. Priority levels can be set with ICR.

5.6.1 Interrupt Control Mode 0

In interrupt control mode 0, interrupt requests other than NMI and address breaks are accepted by the ICR and the I bit of the CCR in the CPU. Figure 5.4 shows a flowchart of the interrupt operation.

1. If an interrupt source occurs when the corresponding interrupt enable bit is set to 1, an interrupt request is sent to the interrupt controller.
2. According to the interrupt control level specified in ICR, the interrupt controller holds pending an interrupt request with interrupt control level 1 (priority), and holds pending an interrupt request with interrupt control level 0 (no priority). If several interrupt requests are pending, the interrupt request with the highest priority is accepted according to the priority order. Interrupt handling is requested to the CPU, and other interrupt requests are held pending.
3. If the I bit in CCR is set to 1, only NMI and address break interrupts are accepted by the interrupt controller, and other interrupt requests are held pending. If the I bit is cleared to 0, any interrupt request is accepted.
4. When the CPU accepts an interrupt request, it starts interrupt exception handling after the execution of the current instruction has been completed.
5. The PC and CCR are saved to the stack area by interrupt exception handling. The top of the stack shows the address of the first instruction to be executed after returning from the interrupt handling routine.

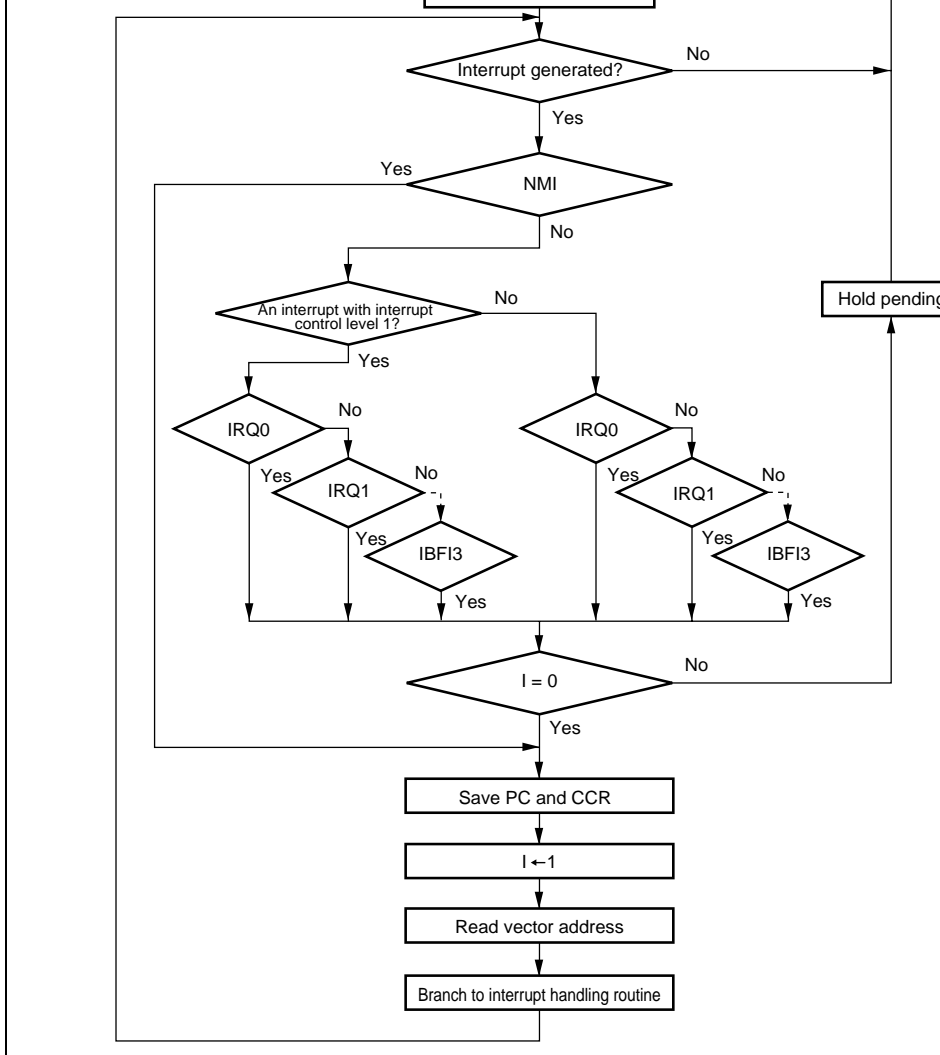


Figure 5.4 Flowchart of Procedure up to Interrupt Acceptance in Interrupt Control

For instance, the state transition when the interrupt enable bit corresponding to each interrupt is set to 1, and ICRA to ICRC are set to H'20, H'00, and H'00, respectively (IRQ2 and IRQ3 are set to control level 1, and other interrupts are set to control level 0) is shown below. Figure 5.5 shows a state transition diagram.

- All interrupt requests are accepted when $I = 0$. (Priority order: NMI > IRQ2 > IRQ3 > address break > IRQ0 > IRQ1 ...)
- Only NMI, IRQ2, IRQ3 and address break interrupt requests are accepted when $I = 1$ and $UI = 0$.
- Only an NMI and address break interrupt request is accepted when $I = 1$ and $UI = 1$.

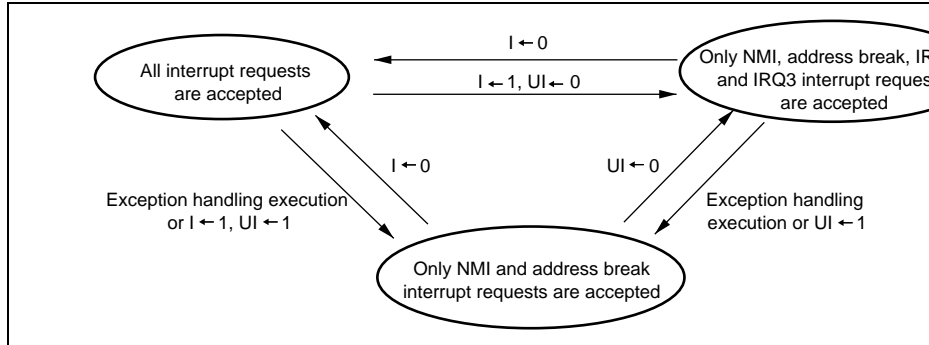


Figure 5.5 State Transition in Interrupt Control Mode 1

- interrupt handling is requested to the CPU, and other interrupt requests are held pending.
3. An interrupt request with interrupt control level 1 is accepted when the I bit is cleared to 0. When the I bit is set to 1 while the UI bit is cleared to 0.
An interrupt request with interrupt control level 0 is accepted when the I bit is cleared to 0. When the I bit is set to 1, only an NMI or address break interrupt request is accepted. Other interrupts are held pending.
When both the I and UI bits are set to 1, only an NMI or address break interrupt request is accepted, and other interrupts are held pending.
When the I bit is cleared to 0, the UI bit is not affected.
 4. When the CPU accepts an interrupt request, it starts interrupt exception handling after the execution of the current instruction has been completed.
 5. The PC and CCR are saved to the stack area by interrupt exception handling. The PC on the stack shows the address of the first instruction to be executed after returning from the interrupt handling routine.
 6. The I and UI bits in CCR are set to 1. This masks all interrupts except for an NMI or address break interrupt.
 7. The CPU generates a vector address for the accepted interrupt and starts execution of the interrupt handling routine at the address indicated by the contents of the vector address table.

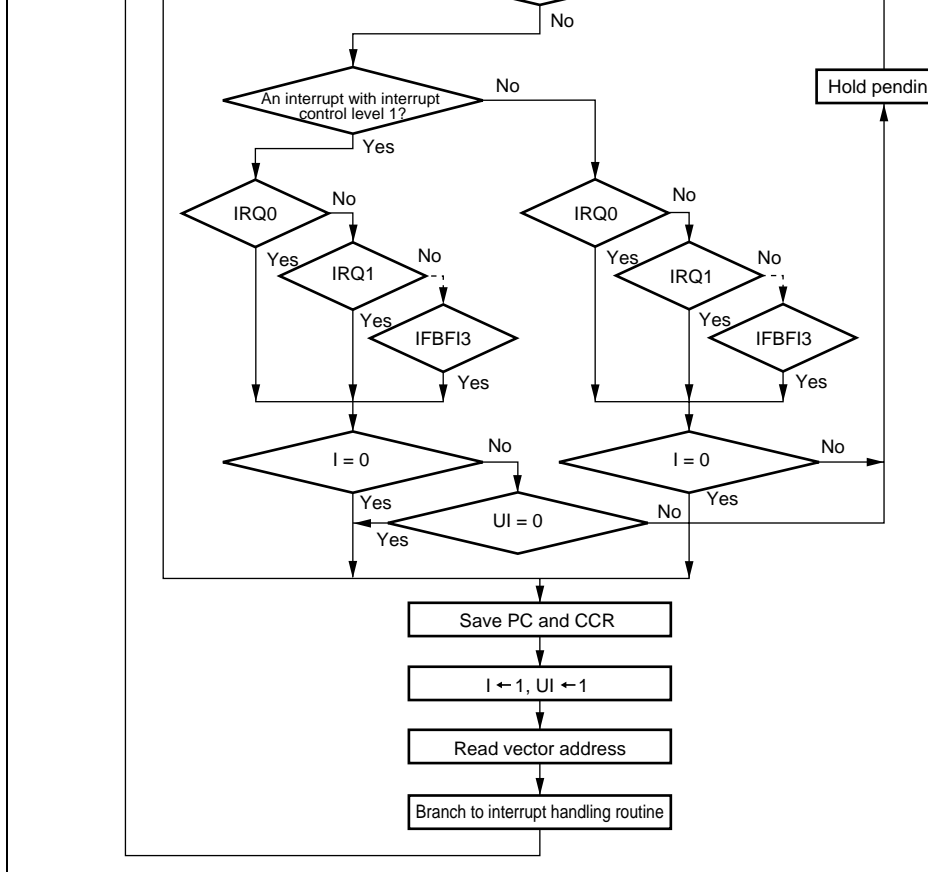


Figure 5.6 Flowchart of Procedure up to Interrupt Acceptance in Interrupt Control Mode 1

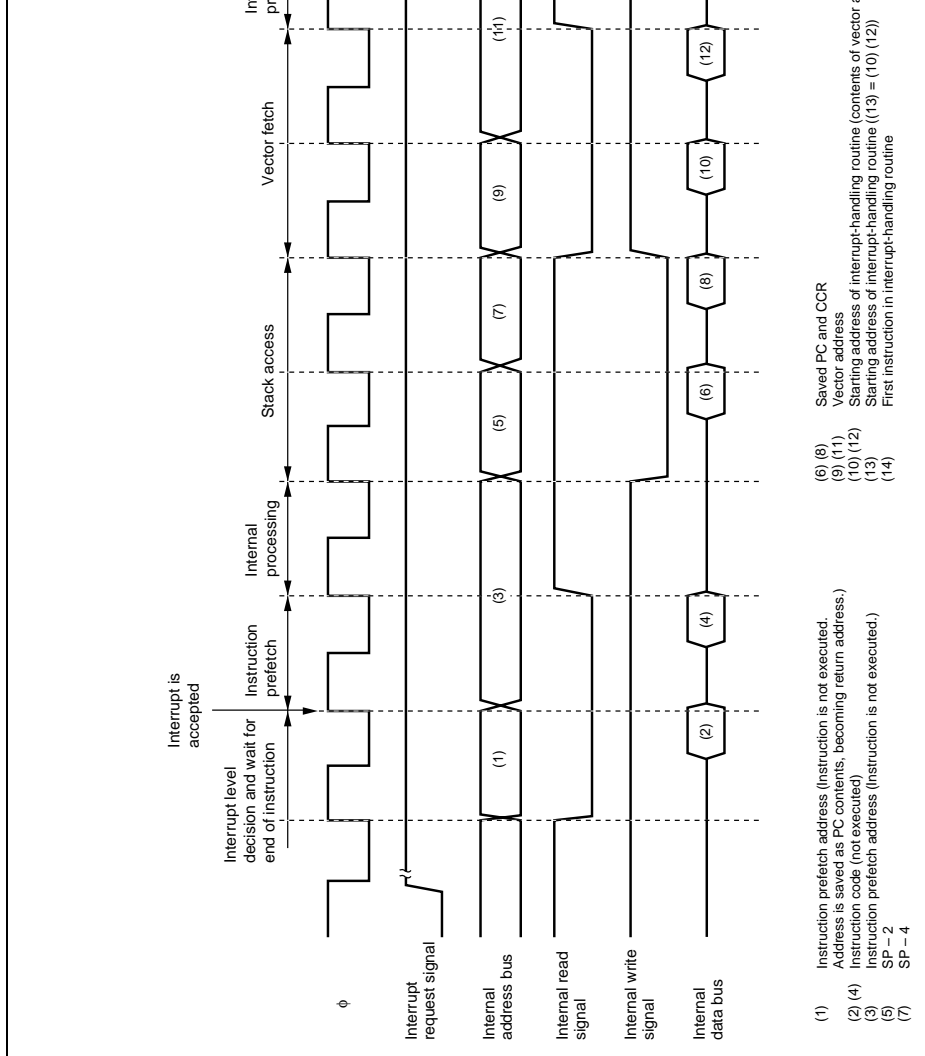


Figure 5.7 Interrupt Exception Handling

1	Interrupt priority determination ^{*1}	3	3
2	Number of wait states until executing instruction ends ^{*2}	1 to (19 + 2·S _I)	1 to (19 + 2·S _I)
3	PC, CCR stack save	2·S _K	2·S _K
4	Vector fetch	S _I	2·S _I
5	Instruction fetch ^{*3}	2·S _I	2·S _I
6	Internal processing ^{*4}	2	2
Total (using on-chip memory)		11 to 31	12 to 32

- Notes: 1. Two states in case of internal interrupt.
2. Refers to MULXS and DIVXS instructions.
3. Prefetch after interrupt acceptance and prefetch of interrupt handling routine.
4. Internal processing after interrupt acceptance and internal processing after v

Table 5.6 Number of States in Interrupt Handling Routine Execution Status

Symbol		Object of Access			
		Internal Memory	External Device		
			8-Bit Bus	16-Bit Bus	16-Bit Bus
		2-State Access	3-State Access	2-State Access	2-State Access
Instruction fetch	S _I	1	4	6 + 2m	2
Branch address read	S _J				
Stack manipulation	S _K				

Legend:

m: Number of wait states in external device access

For details on interrupt requests that can be used to activate the DTC, see section 7, DTC Controller (DTC).

Figure 5.8 shows a block diagram of the DTC and interrupt controller.

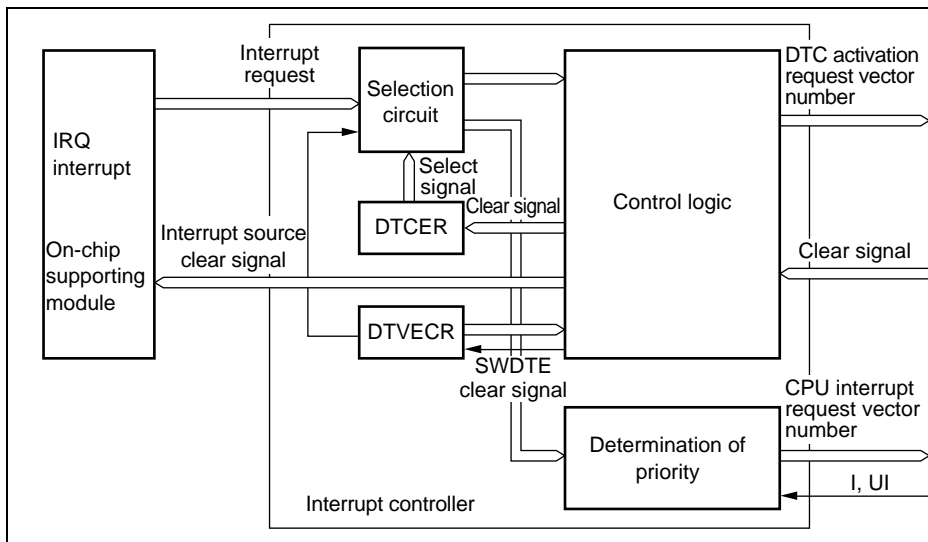


Figure 5.8 DTC and Interrupt Controller

Selection of Interrupt Source: Interrupt factors are selected as DTC activation source interrupt source by the DTCE bit of DTCERA to DTCERE of DTC.

By specifying the DISEL bit of the DTC's MRB, it is possible to clear the DTCE bit to stop DTC data transfer, and request a CPU interrupt.

If DTC carries out the designate number of data transfers and the transfer counter reaches the DTVECR, the DTCE bit is also cleared to 0, and an interrupt is requested to the CPU.

of the DTCE bit of DTC's DTCER, and the DISEL bit of DTC's MRB.

Table 5.7 Interrupt Source Selection and Clearing Control

Settings		Interrupt Sources Selection/Clearing	
DTCE	DTC	DTC	CPU
	DISEL		
0	*	×	○
1	0	○	×
	1	○	○

Legend:

○: The relevant interrupt is used. Interrupt source clearing is performed.

(The CPU should clear the source flag in the interrupt handling routine.)

○: The relevant interrupt is used. The interrupt source is not cleared.

×: The relevant interrupt cannot be used.

*: Don't care

Note: The SCI, IIC, LPC, or A/D converter interrupt source is cleared when the DTC writes to the prescribed register, and is not dependent upon the DISEL bit.

With this function, the execution start point of a program containing a bug is detected and execution is branched to the correcting program.

5.7.2 Block Diagram

Figure 5.9 shows a block diagram of the address break.

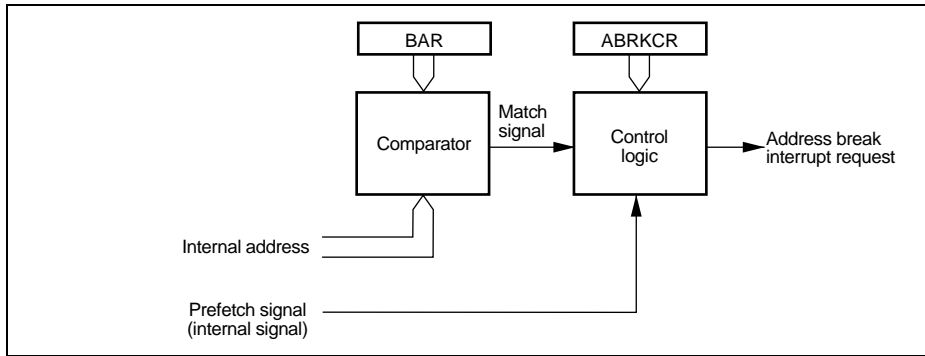


Figure 5.9 Address Break Block Diagram

To use the address break function, set each register as follows:

1. Set a break address in the A23 to A1 bits in BAR.
2. Set the BIE bit in ABRKCR to 1 to enable the address break.

When the BIE bit is cleared to 0, an address break is not requested.

When the setting conditions are satisfied, the CMF flag in ABRKCR is set to 1 to request an interrupt. The interrupt source should be determined by the interrupt handling routine in the user program.

5.7.4 Usage Notes

1. In an address break, the break address should be an address where the first byte of the instruction exists. Otherwise, a break condition will not be satisfied.
2. In normal mode, addresses A23 to A16 are not compared.
3. When the branch instructions (Bcc, BSR), jump instructions (JMP, JSR), RST instruction, and RTE instruction are placed immediately prior to the address specified by BAR, a prefetch signal to the address may be output to request an address break by executing these instructions. It is necessary to take countermeasures: do not set a break address to an address immediately after these instructions, or determine whether interrupt handling is performed by satisfying a normal condition.
4. An address break interrupt is generated by combining the internal prefetch signal and the address. Therefore, the timing to enter the interrupt exception handling differs according to the instructions at the specified and at prior addresses and execution cycles.

Figure 5.10 shows an example of address timing.

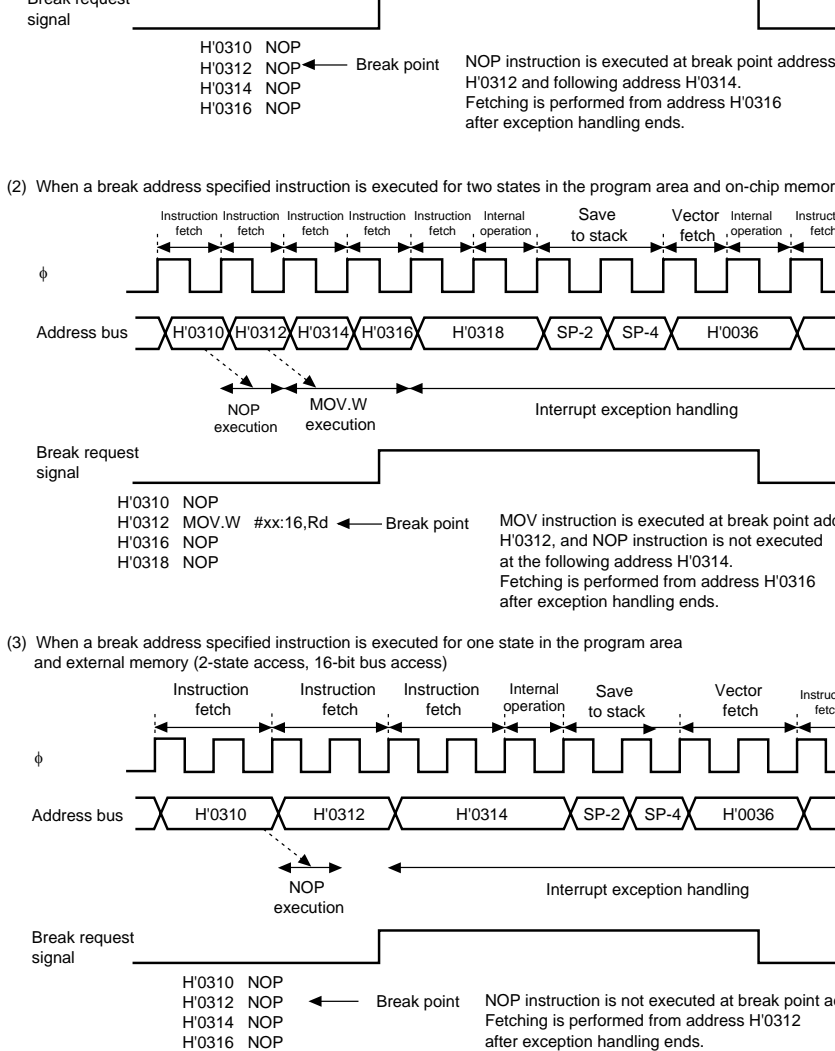


Figure 5.10 Address Break Timing Example

interrupt exception handling for that interrupt will be executed on completion of the ins
 However, if there is an interrupt request of higher priority than that interrupt, interrupt
 handling will be executed for the higher-priority interrupt, and the lower-priority interr
 ignored. The same rule is also applied when an interrupt source flag is cleared to 0. Fig
 shows an example in which the CMIEA bit in the TMR's TCR register is cleared to 0.

The above conflict will not occur if an enable bit or interrupt source flag is cleared to 0.
 interrupt is masked.

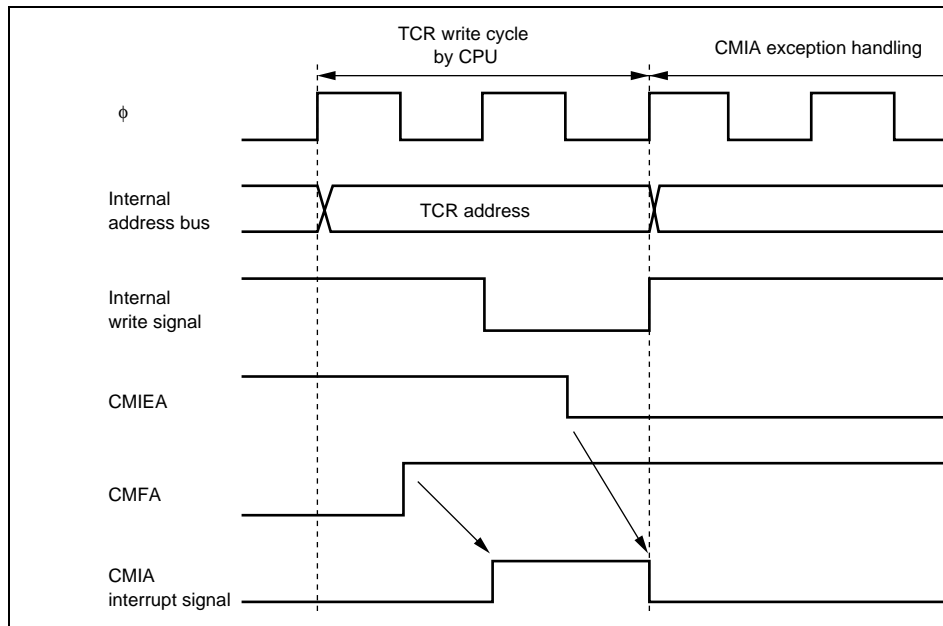


Figure 5.11 Conflict between Interrupt Generation and Disabling

Interrupt operation differs between the EEPMOV.B instruction and the EEPMOV.W instruction.

With the EEPMOV.B instruction, an interrupt request (including NMI) issued during the transfer is not accepted until the move is completed.

With the EEPMOV.W instruction, if an interrupt request is issued during the transfer, exception handling starts at a break in the transfer cycle. The PC value saved on the stack in this case is the address of the next instruction. Therefore, if an interrupt is generated during the transfer of an EEPMOV.W instruction, the following coding should be used.

```
L1 :      EEPMOV.W
          MOV.W    R4, R4
          BNE     L1
```

5.8.4 Setting on Product Incorporating DTC

When a product, in which a DTC is incorporated, is used in the following settings, the corresponding flag bit is not automatically cleared even when exception handling, which is executed under the condition, is executed and the bit is held at 1.

1. When DTCEA3 is set to 1 (ADI is set to an interrupt source), IRQ4F flag is not automatically cleared.
2. When DTCEA2 is set to 1 (ICIA is set to an interrupt source), IRQ5F flag is not automatically cleared.
3. When DTCEA1 is set to 1 (ICIB is set to an interrupt source), IRQ6F flag is not automatically cleared.
4. When DTCEA0 is set to 1 (OCIA is set to an interrupt source), IRQ7F flag is not automatically cleared.

When activation interrupt sources of DTC and IRQ interrupts are used with the above combinations, clear the interrupt flag by software in the interrupt handling routine of the corresponding IRQ.

- **Basic bus interface**
2-state access or 3-state access can be selected for each area
Program wait states can be inserted for each area
- **Burst ROM interface**
A burst ROM interface can be set for basic expansion areas
1-state access or 2-state access can be selected for burst access
- **Idle cycle insertion**
An idle cycle can be inserted for external write cycles immediately after external r
- **Bus arbitration function**
Includes a bus arbiter that arbitrates bus mastership between the CPU and DTC

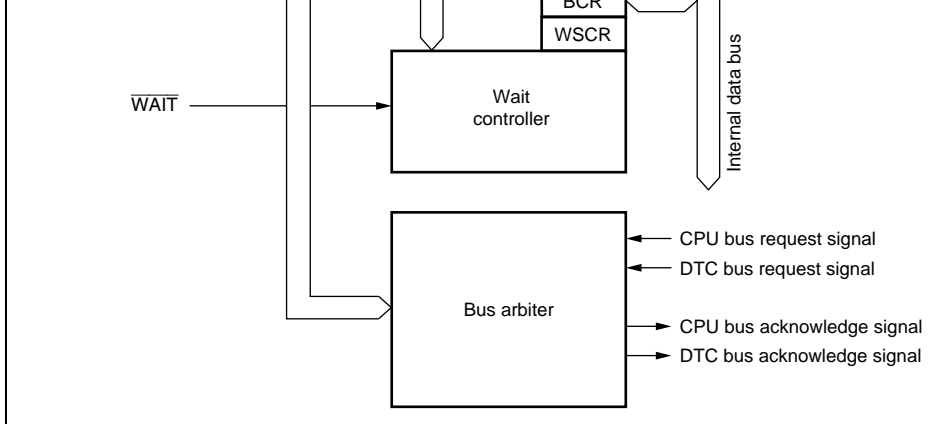


Figure 6.1 Block Diagram of Bus Controller

$\overline{\text{IOS}}$	Output	I/O select signal (when the IOSE bit in SYSCR is set).
$\overline{\text{RD}}$	Output	Strobe signal indicating that the external address bus is being read.
$\overline{\text{HWR}}$	Output	Strobe signal indicating that the external address bus is being written to, and the upper half (D15 to D8) of the bus is enabled.
$\overline{\text{LWR}}$	Output	Strobe signal indicating that the external address bus is being written to, and the lower half (D7 to D0) of the bus is enabled.
$\overline{\text{WAIT}}$	Input	Wait request signal when accessing the external access space.

6.3 Register Descriptions

The bus controller has the following registers. For details on the system control registers, see section 3.2.2, System Control Register (SYSCR).

- Bus control register (BCR)
- Wait state control register (WSCR)

6	ICIS0	1	R/W	<p>Idle Cycle Insertion</p> <p>Selects whether or not to insert 1-state of the idle cycle between bus cycles when the external read cycle follows the external read cycle.</p> <p>0: Idle cycle not inserted when the external read cycle follows the external read cycle</p> <p>1: 1-state idle cycle inserted when the external read cycle follows the external read cycle</p>
5	BRSTRM	0	R/W	<p>Burst ROM Enable</p> <p>Selects the bus interface for the external address space.</p> <p>0: Basic bus interface</p> <p>1: Burst ROM interface</p>
4	BRSTS1	1	R/W	<p>Burst Cycle Select 1</p> <p>Selects the number of states in the burst cycle of the burst ROM interface.</p> <p>0: 1 state</p> <p>1: 2 states</p>
3	BRSTS0	0	R/W	<p>Burst Cycle Select 0</p> <p>Selects the number of words that can be accessed in a burst access via the burst ROM interface.</p> <p>0: Max, 4 words</p> <p>1: Max, 8 words</p>
2	—	0	R/W	<p>Reserved</p> <p>This bit should not be written by 0.</p>
1	IOS1	1	R/W	IOS Select 1, 0
0	IOS0	1	R/W	Select the address range where the $\overline{\text{IOS}}$ signal is active. For details, refer to table 6.3.

7, 6			R/W	Reserved	These bits should not be written by 1.
5	ABW	1	R/W	Bus Width Control	<p>Selects 8 or 16 bits for access to the external address space.</p> <p>0: 16-bit access space</p> <p>1: 8-bit access space</p>
4	AST	1	R/W	Access State Control	<p>Selects 2 or 3 access states for access to the external address space. This bit also enables or disables wait state insertion.</p> <p>0: 2-state access space. Wait state insertion disabled in external address space access</p> <p>1: 3-state access space. Wait state insertion enabled in external address space access</p>
3	WMS1	0	R/W	Wait Mode Select 1, 0	
2	WMS0	0	R/W	Select the wait mode for access to the external address space when the AST bit is set to 1	<p>00: Program wait mode</p> <p>01: Wait disabled mode</p> <p>10: Pin wait mode</p> <p>11: Pin auto-wait mode</p>

6.4 Bus Control

6.4.1 Bus Specifications

The external address space bus specifications consist of three elements: Bus width, the access states, and the wait mode and the number of program wait states. The bus width number of access states for on-chip memory and internal I/O registers are fixed, and are affected by the bus controller settings.

Bus Width: A bus width of 8 or 16 bits can be selected via the ABW bit in WSCR.

Number of Access States: Two or three access states can be selected via the AST bit in WSCR. When the 2-state access space is designated, wait-state insertion is disabled.

In the burst ROM interface, the number of access states is determined regardless of the setting.

Wait Mode and Number of Program Wait States: When a 3-state access space is designated via the AST bit in WSCR, the wait mode and the number of program wait states to be inserted automatically is selected by the WMS1, WMS0, WC1, and WC0 bits in WSCR. From 0 to 3 program wait states can be selected.

1	0	1	—	—	16	3	0
	—*	—*	0	0		3	0
				1			1
			1	0			2
				1			3
1	0	—	—	—	8	2	0
	1	0	1	—	8	3	0
		—*	—*	0	0	3	0
					1		1
				1	0		2
					1		3

Note: * Other than WMS1 = 0 and WMS0 = 1

6.4.2 Advanced Mode

The external address space is initialized as the basic bus interface and a 3-state access on-chip ROM enable extended mode, the address space other than on-chip ROM, on-chip RAM and its reserved area are enabled when the RAME bit in SYSCR is set to 1. In on-chip ROM disable extended mode, the address space other than on-chip ROM, on-chip RAM, internal I/O registers, and their reserved areas is specified as the external address space. The on-chip ROM area is disabled and corresponding addresses are the external address space when the RAME bit is cleared to 0.

6.4.3 Normal Mode

The external address space is initialized as the basic bus interface and a 3-state access on-chip ROM disable extended mode, the address space other than on-chip ROM, on-chip RAM and its reserved area are disabled and corresponding addresses are the external address space. In on-chip ROM enable extended mode, the address space other than on-chip ROM, on-chip RAM, internal I/O registers, and their reserved areas is specified as the external address space. The on-chip RAM area is enabled when the RAME bit is set to 1.

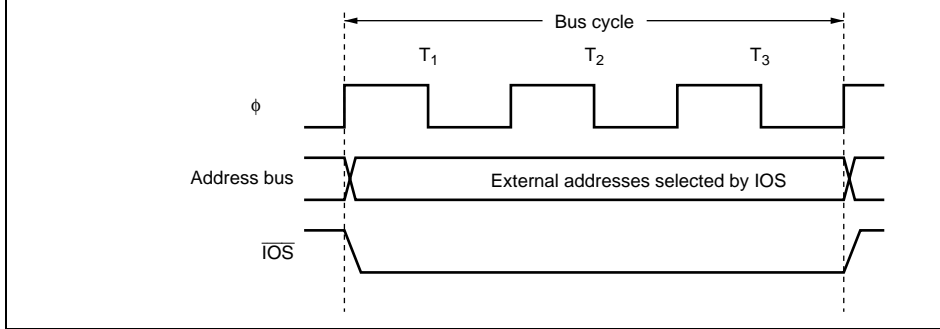


Figure 6.2 $\overline{\text{IOS}}$ Signal Output Timing

Enabling or disabling $\overline{\text{IOS}}$ signal output is performed by the IOSE bit in SYSCR. In external mode, the $\overline{\text{IOS}}$ pin functions as an $\overline{\text{AS}}$ pin by a reset. To use this pin as an $\overline{\text{IOS}}$ pin, set the IOSE bit to 1. For details, refer to section 8, I/O Ports.

The address ranges of the $\overline{\text{IOS}}$ signal output can be specified by the IOS1 and IOS0 bits as shown in table 6.3.

Table 6.3 Address Range for $\overline{\text{IOS}}$ Signal Output

IOS1	IOS0	$\overline{\text{IOS}}$ Signal Output Range
0	0	H'(FF)F000 to H'(FF)F03F
	1	H'(FF)F000 to H'(FF)F0FF
1	0	H'(FF)F000 to H'(FF)F3FF
	1	H'(FF)F000 to H'(FF)F7FF

a data alignment function, and controls whether the upper data bus (D15 to D8) or lower data bus (D7 to D0) is used when the external address space is accessed, according to the bus size for the area being accessed (8-bit access space or 16-bit access space) and the data size.

8-Bit Access Space: Figure 6.3 illustrates data alignment control for the 8-bit access space. In the 8-bit access space, the upper data bus (D15 to D8) is always used for accesses. The amount of data that can be accessed at one time is one byte: a word access is performed as two byte accesses, and a longword access, as four byte accesses.

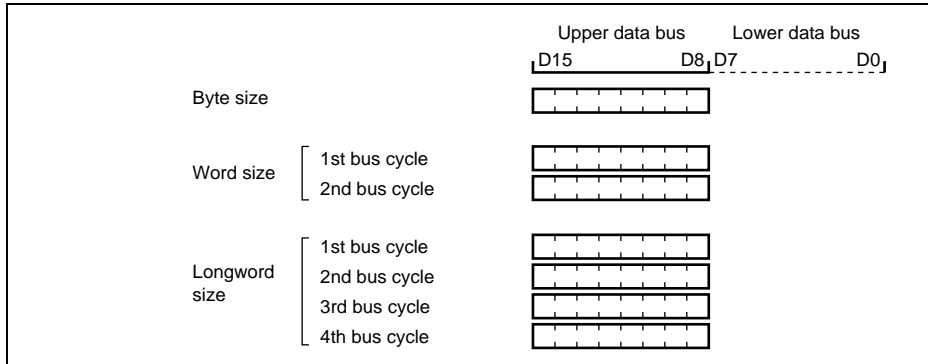


Figure 6.3 Access Sizes and Data Alignment Control (8-Bit Access Space)

16-Bit Access Space: Figure 6.4 illustrates data alignment control for the 16-bit access space. With the 16-bit access space, the upper data bus (D15 to D8) and lower data bus (D7 to D0) are used for accesses. The amount of data that can be accessed at one time is one byte or one word, and a longword access is executed as two word accesses.

In byte access, whether the upper or lower data bus is used is determined by whether the address is even or odd. The upper data bus is used for an even address, and the lower data bus is used for an odd address.

6.5.2 Valid Strobes

Table 6.4 shows the data buses used and valid strobes for each access space.

In a read, the \overline{RD} signal is valid for both the upper and lower halves of the data bus. In a write, the \overline{HWR} signal is valid for the upper half of the data bus, and the \overline{LWR} signal for the lower half.

Table 6.4 Data Buses Used and Valid Strobes

Area	Access Size	Read/Write	Address	Valid Strobe	Upper Data Bus (D15 to D8)	Lower Data Bus (D7 to D0)
8-bit access space	Byte	Read	—	\overline{RD}	Valid	Ports or others
		Write	—	\overline{HWR}		Ports or others
16-bit access space	Byte	Read	Even	\overline{RD}	Valid	Invalid
			Odd		Invalid	Valid
		Write	Even	\overline{HWR}	Valid	Undefined
			Odd	\overline{LWR}	Undefined	Valid
	Word	Read	—	\overline{RD}	Valid	Valid
		Write	—		$\overline{HWR}, \overline{LWR}$	Valid

Note: Undefined: Undefined data is output.
 Invalid: Input state with the input value ignored.
 Ports or others: Used as ports or I/O pins for on-chip peripheral modules, and as the data bus.

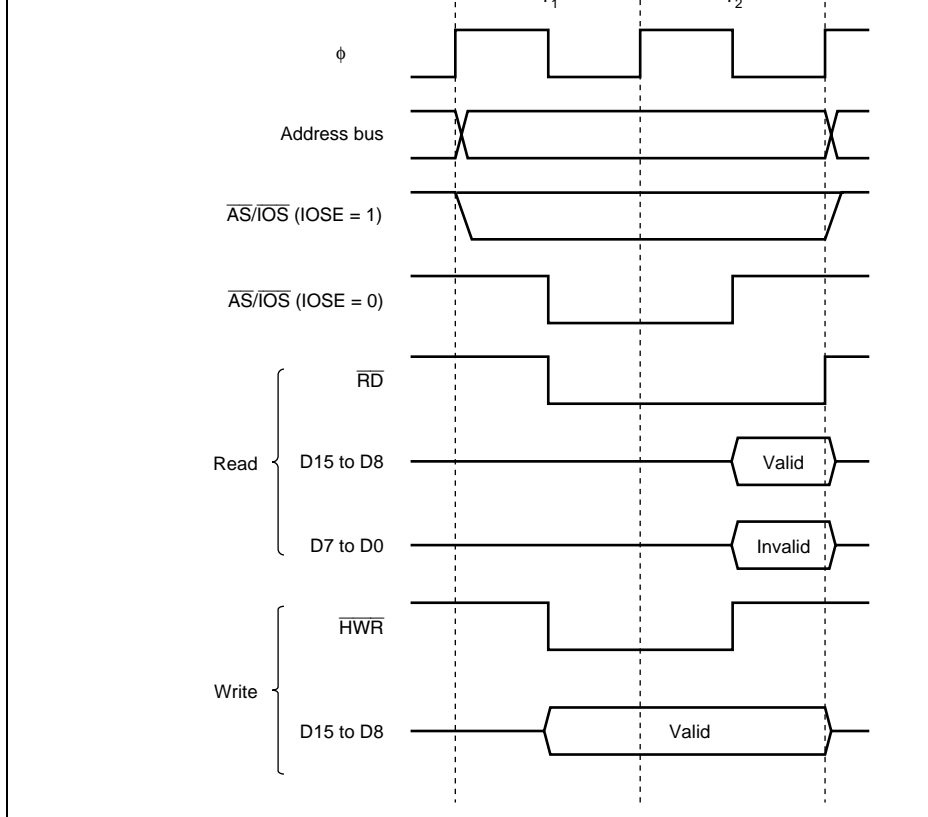


Figure 6.5 Bus Timing for 8-Bit, 2-State Access Space

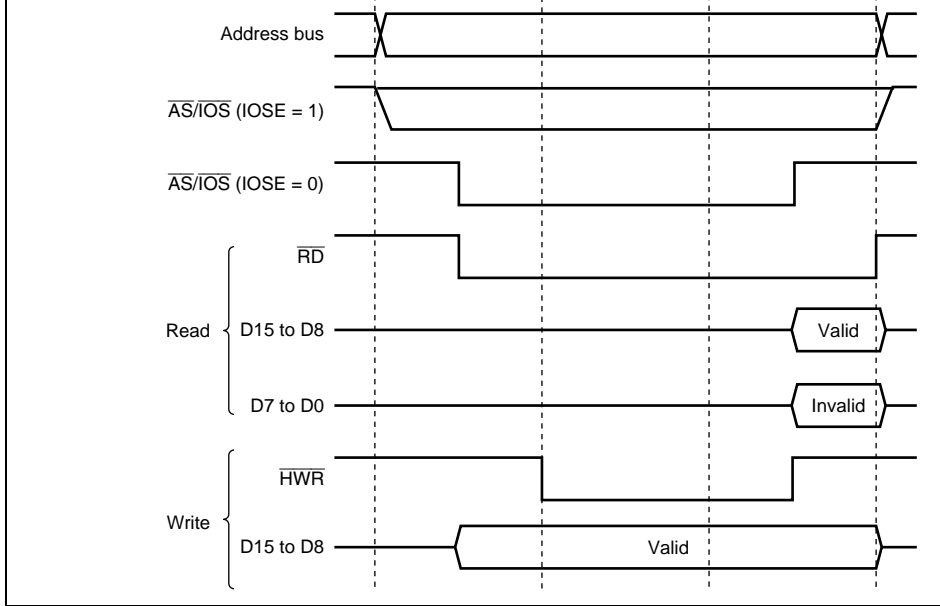


Figure 6.6 Bus Timing for 8-Bit, 3-State Access Space

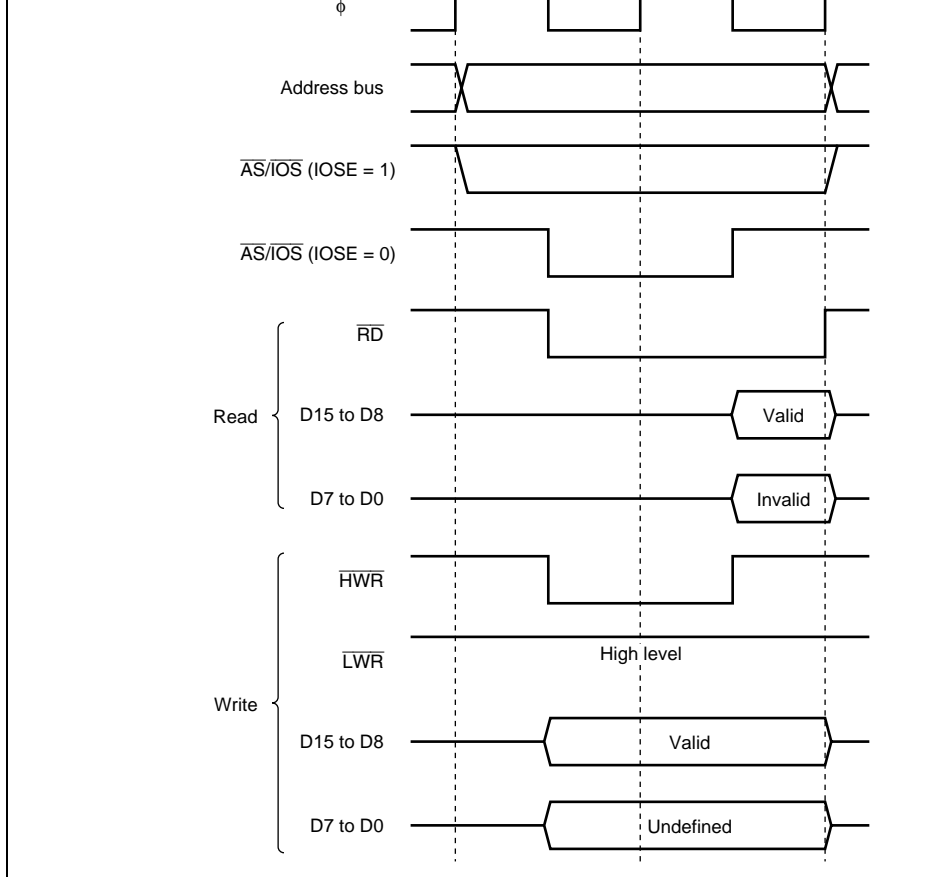


Figure 6.7 Bus Timing for 16-Bit, 2-State Access Space (Even Byte Access)

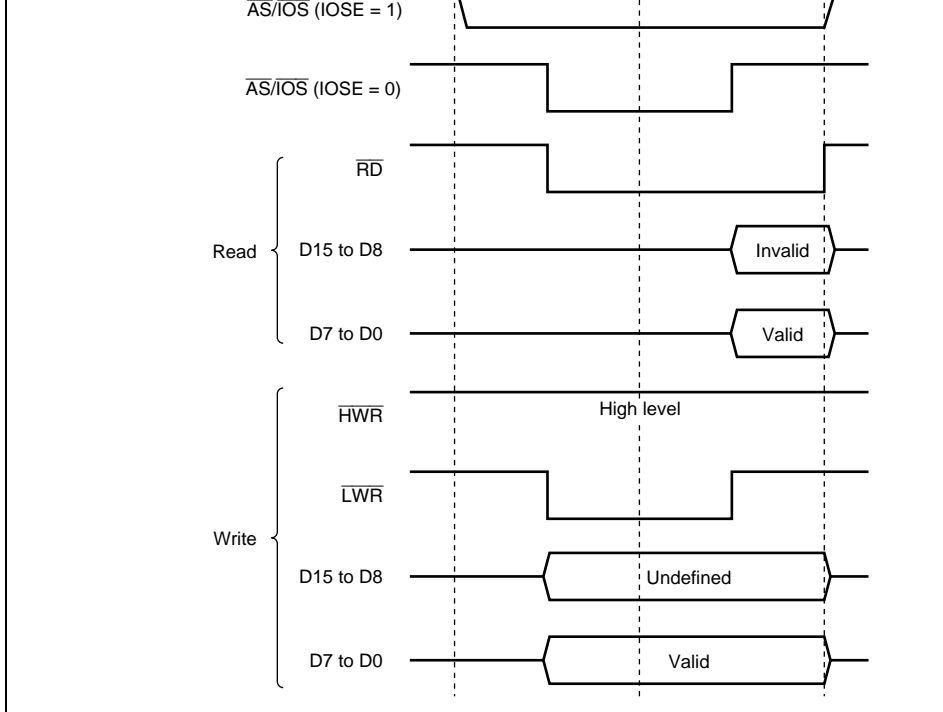


Figure 6.8 Bus Timing for 16-Bit, 2-State Access Space (Odd Byte Access)

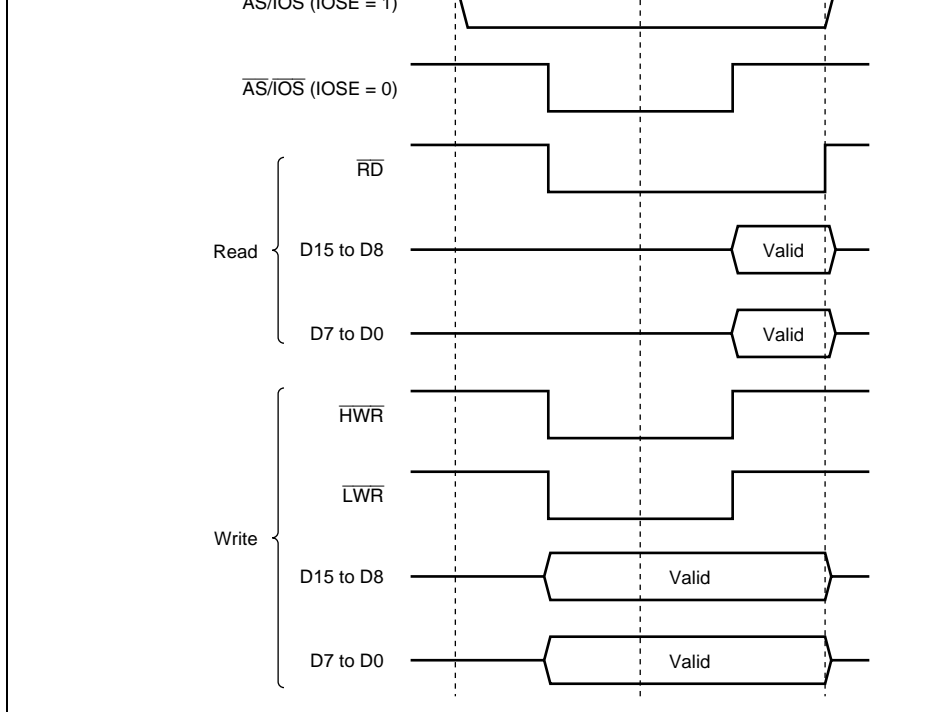


Figure 6.9 Bus Timing for 16-Bit, 2-State Access Space (Word Access)

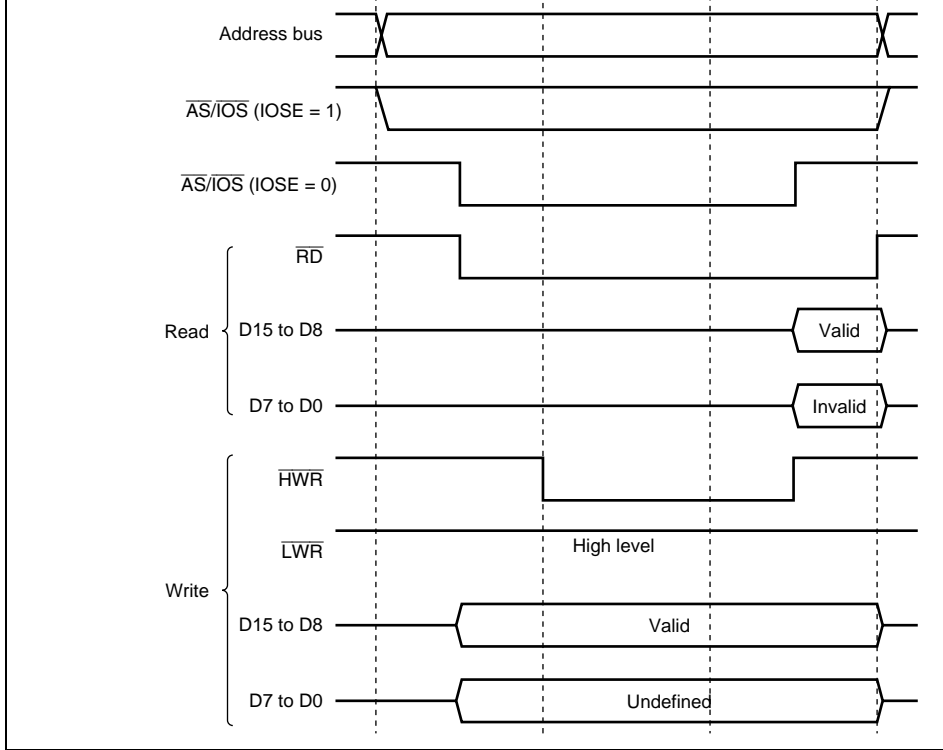


Figure 6.10 Bus Timing for 16-Bit, 3-State Access Space (Even Byte Access)

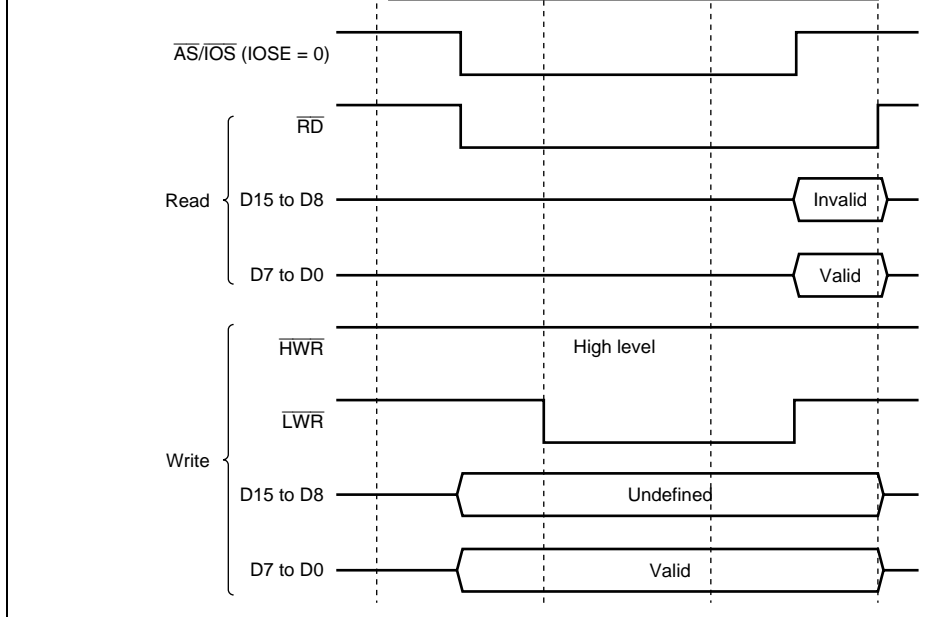


Figure 6.11 Bus Timing for 16-Bit, 3-State Access Space (Odd Byte Access)

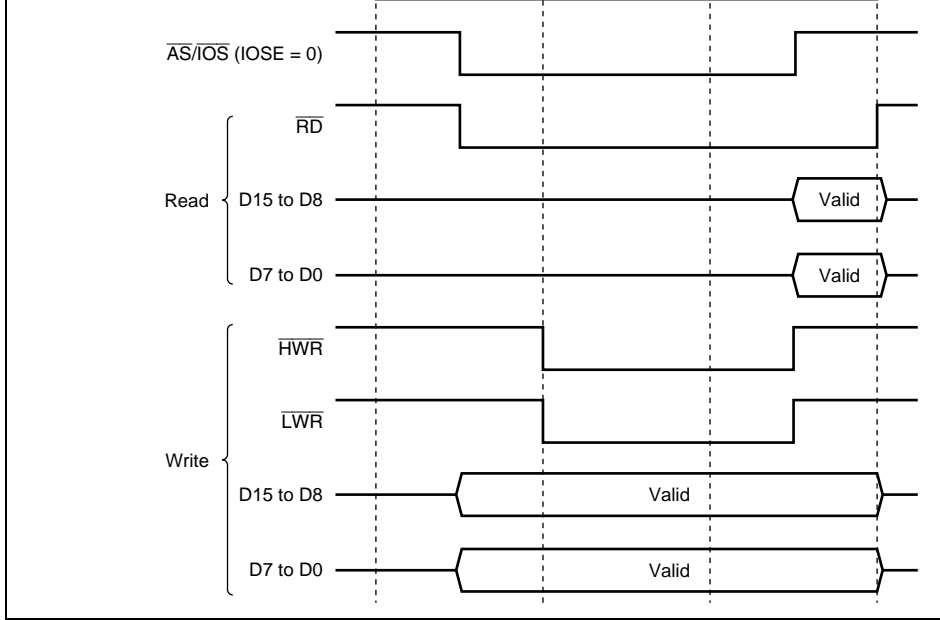


Figure 6.12 Bus Timing for 16-Bit, 3-State Access Space (Word Access)

the settings of the WC1 and WC0 bits in WSCR.

Pin Wait Mode: A specified number of wait states T_w can be inserted automatically between the T_2 state and T_3 state when accessing the external address space always according to the settings of the WC1 and WC0 bits. If the $\overline{\text{WAIT}}$ pin is low at the falling edge of ϕ in the last T_2 or another T_w state is inserted. If the $\overline{\text{WAIT}}$ pin is held low, T_w states are inserted until it

This is useful when inserting four or more T_w states, or when changing the number of wait states to be inserted for each external device.

Pin Auto-Wait Mode: A specified number of wait states T_w can be inserted automatically between the T_2 state and T_3 state when accessing the external address space according to the settings of the WC1 and WC0 bits if the $\overline{\text{WAIT}}$ pin is low at the falling edge of ϕ in the last T_2 state. Even if the $\overline{\text{WAIT}}$ pin is held low, T_w states can be inserted only up to the specified number of states.

This function enables the low-speed memory interface only by inputting the chip select to the $\overline{\text{WAIT}}$ pin.

Figure 6.13 shows an example of wait state insertion timing in pin wait mode.

The settings after a reset are: 3-state access, 3 program wait insertion, and $\overline{\text{WAIT}}$ pin input disabled.

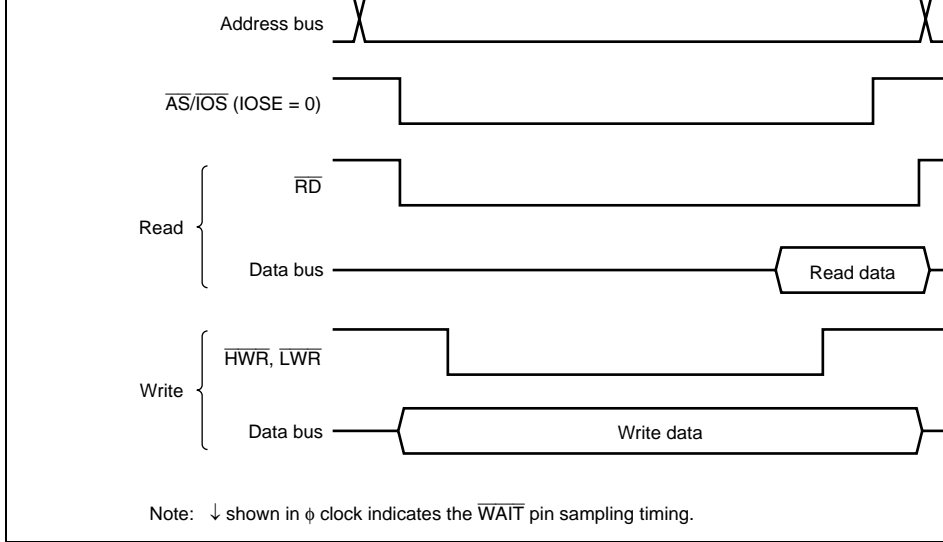


Figure 6.13 Example of Wait State Insertion Timing (Pin Wait Mode)

The number of access states in the initial cycle (full access) of the burst ROM interface determined by the AST bit in WSCR. When the AST bit is set to 1, wait states can be inserted or 2 states can be selected for burst access according to the setting of the BRSTS1 bit. Wait states cannot be inserted in a burst cycle. Burst accesses of a maximum four words are performed when the BRSTS0 bit in BCR is cleared to 0, and burst accesses of a maximum of 16 words is performed when the BRSTS0 bit in BCR is set to 1.

The basic access timing for the burst ROM space is shown in figures 6.14 and 6.15.

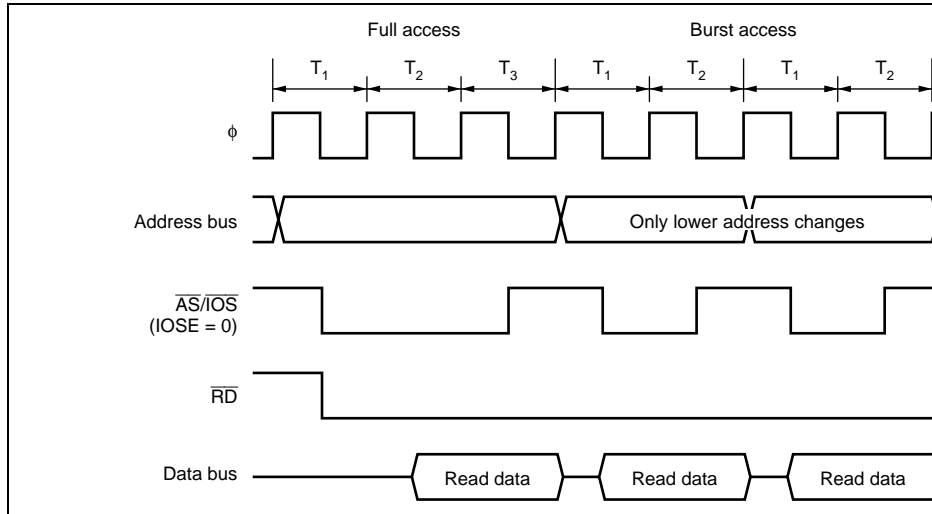


Figure 6.14 Access Timing Example in Burst ROM Space (AST = BRSTS0)

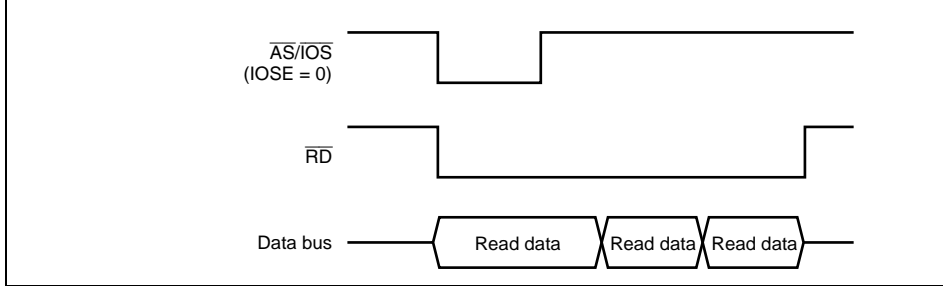


Figure 6.15 Access Timing Example in Burst ROM Space (AST = BRSTS1)

6.6.2 Wait Control

As with the basic bus interface, program wait insertion or pin wait insertion using the \overline{V} can be used in the initial cycle (full access) of the burst ROM interface. For details, see 6.5.4, Wait Control. Wait states cannot be inserted in a burst cycle.

6.7 Idle Cycle

When this LSI accesses the external address space, it can insert a 1-state idle cycle (T_{1}) bus cycles when a write cycle occurs immediately after a read cycle. By inserting an idle cycle, if possible, for example, to avoid data collisions between ROM with a long output floating time and high-speed memory and I/O interfaces.

If an external write occurs after an external read while the ICIS0 bit is set to 1 in BCR, an idle cycle is inserted at the start of the write cycle.

Figure 6.16 shows examples of idle cycle operation. In these examples, bus cycle A is a read cycle for ROM with a long output floating time, and bus cycle B is a CPU write cycle. In figure 6.16 (a), with no idle cycle inserted, a collision occurs in bus cycle B between the read data from the ROM and the CPU write data. In figure 6.16 (b), an idle cycle is inserted, thus preventing data collisions.

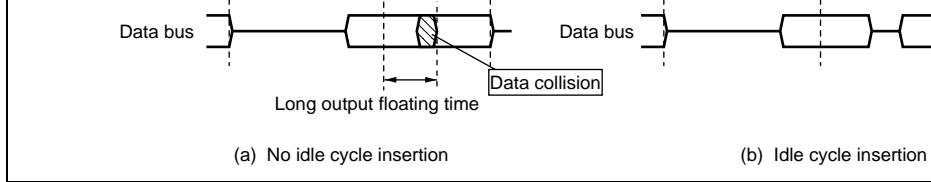


Figure 6.16 Examples of Idle Cycle Operation

Table 6.5 shows the pin states in an idle cycle.

Table 6.5 Pin States in Idle Cycle

Pins	Pin State
A23 to A0, \overline{IOS}	Contents of immediately following bus cycle
D15 to D0	High impedance
\overline{AS}	High
\overline{RD}	High
\overline{HWR} , \overline{LWR}	High

Each bus master requests the bus by means of a bus request signal. The bus arbiter detects the masters' bus request signals, and if a bus request occurs, it sends a bus request acknowledge signal to the bus master making the request at the designated timing. If there are bus requests from more than one bus master, the bus request acknowledge signal is sent to the one with the highest priority. When a bus master receives the bus request acknowledge signal, it takes possession of the bus until that signal is canceled. The order of priority of the bus masters is as follows:

(High) DTC > CPU (Low)

6.8.2 Bus Transfer Timing

When a bus request is received from a bus master with a higher priority than that of the bus master that has acquired the bus and is currently operating, the bus is not necessarily transferred immediately. Each bus master can relinquish the bus at the timings given below.

CPU: The CPU is the lowest-priority bus master, and if a bus request is received from a bus master with a higher priority than the bus arbiter transfers the bus to the DTC.

- DTC bus transfer timing
 - The bus is transferred at a break between bus cycles. However, if a bus cycle is used for discrete operations, as in the case of a longword-size access, the bus is not transferred between the component operations. For details, refer to the H8S/2600 Series, H8C/2600 Series Programming Manual.
 - If the CPU is in sleep mode, the bus is transferred immediately.

DTC: The DTC has the highest bus master priority. The DTC sends the bus arbiter a request for the bus when an activation request is generated. The DTC does not release the bus until it completes its operation.

7.1 Features

- Transfer is possible over any number of channels
- Three transfer modes
Normal, repeat, and block transfer modes are available.
- One activation source can trigger a number of data transfers (chain transfer)
- Direct specification of 16-Mbyte address space is possible
- Activation by software is possible
- Transfer can be set in byte or word units
- A CPU interrupt can be requested for the interrupt that activated the DTC
- Module stop mode can be set

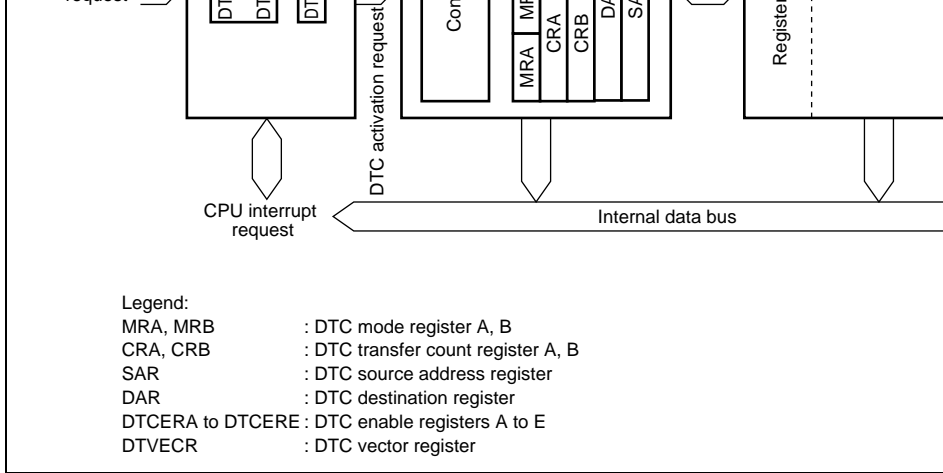


Figure 7.1 Block Diagram of DTC

- DTC transfer count register A (CRA)
- DTC transfer count register B (CRB)

These six registers cannot be directly accessed from the CPU. When a DTC activation source occurs, the DTC reads a set of register information that is stored in on-chip RAM corresponding DTC registers and transfers data. After the data transfer, it writes a set of register information back to on-chip RAM.

- DTC enable registers A to E (DTCERA to DTCERE)
- DTC vector register (DTVECR)

7.2.1 DTC Mode Register A (MRA)

MRA selects the DTC operating mode.

(by -1 when Sz = 0, by -2 when Sz = 1)				
5	DM1	Undefined	—	Destination Address Mode 1, 0
4	DM0	Undefined	—	<p>These bits specify a DAR operation after a d transfer.</p> <p>0X: DAR is fixed</p> <p>10: DAR is incremented after a transfer (by +1 when Sz = 0, by +2 when Sz = 1)</p> <p>11: DAR is decremented after a transfer (by -1 when Sz = 0, by -2 when Sz = 1)</p>
3	MD1	Undefined	—	DTC Mode
2	MD0	Undefined	—	<p>These bits specify the DTC transfer mode.</p> <p>00: Normal mode</p> <p>01: Repeat mode</p> <p>10: Block transfer mode</p> <p>11: Setting prohibited</p>
1	DTS	Undefined	—	<p>DTC Transfer Mode Select</p> <p>Specifies whether the source side or the des side is set to be a repeat area or block area mode or block transfer mode.</p> <p>0: Destination side is repeat area or block ar</p> <p>1: Source side is repeat area or block area</p>
0	Sz	Undefined	—	<p>DTC Data Transfer Size</p> <p>Specifies the size of data to be transferred.</p> <p>0: Byte-size transfer</p> <p>1: Word-size transfer</p>

Legend:

X: Don't care

In data transfer with CHNE set to 1, determining the end of the specified number of data transfer, clearing of the interrupt source flag, and clearing of the DTCER are not performed.

6	DISEL	Undefined	—	DTC Interrupt Select
<p>When this bit is set to 1, a CPU interrupt request is generated every time data transfer ends (the DTC clears the interrupt source flag for the activation source). When this bit is cleared to 0, a CPU interrupt request is generated only when the specified data transfer ends (the DTC does not clear the interrupt source flag for the activation source).</p>				
5 to 0	—	Undefined	—	Reserved
<p>These bits have no effect on DTC operation. No data should be written to these bits.</p>				

7.2.3 DTC Source Address Register (SAR)

SAR is a 24-bit register that designates the source address of data to be transferred by the DTC. For word-size transfer, specify an even source address.

7.2.4 DTC Destination Address Register (DAR)

DAR is a 24-bit register that designates the destination address of data to be transferred by the DTC. For word-size transfer, specify an even destination address.

functions as an 8-bit transfer counter (1 to 256). CRAL is decremented by 1 every time transferred, and the contents of CRAH are sent when the count reaches H'00.

7.2.6 DTC Transfer Count Register B (CRB)

CRB is a 16-bit register that designates the number of times data is to be transferred by block transfer mode. It functions as a 16-bit transfer counter (1 to 65536) that is decremented every time data is transferred, and transfer ends when the count reaches H'0000.

7.2.7 DTC Enable Registers (DTCER)

DTCER specifies DTC activation interrupt sources. DTCER is comprised of five registers, DTCERA to DTCERE. The correspondence between interrupt sources and DTCE bits is shown in table 7.1. For DTCE bit setting, use bit manipulation instructions such as BSET and BCLR. Multiple DTC activation sources can be set at one time (only at the initial setting) by multiple interrupts and writing data after executing a dummy read on the relevant register.

Bit	Bit Name	Initial Value	R/W	Description
7	DTCE7	0	R/W	DTC Activation Enable
6	DTCE6	0	R/W	Setting this bit to 1 specifies a relevant interrupt source as a DTC activation source.
5	DTCE5	0	R/W	[Clearing conditions]
4	DTCE4	0	R/W	• When data transfer has ended with the DTCMRB set to 1.
3	DTCE3	0	R/W	• When the specified number of transfers has been completed.
2	DTCE2	0	R/W	[Holding condition]
1	DTCE1	0	R/W	When the DISSEL bit is 0 and the specified number of transfers have not been completed.
0	DTCE0	0	R/W	

Setting this bit to 1 activates DTC. Only 1 can be written to this bit. 0 can be written to after request from this bit.

[Clearing conditions]

- When the DISEL bit is 0 and the specified number of transfers have not ended.
- When 0 is written to the DISEL bit after the DTC is activated data transfer end interrupt (SWDTM) request has been sent to the CPU.

[Holding conditions]

- When the DISEL bit is 1 and data transfer has ended
- When the specified number of transfers has ended
- During data transfer activated by software

6	DTVEC6	0	R/W	DTC Software Activation Vectors 6 to 0
5	DTVEC5	0	R/W	These bits specify a vector number for DTC activation.
4	DTVEC4	0	R/W	
3	DTVEC3	0	R/W	The vector address is expressed as H'0400 + (vector number × 2). For example, when DTVEC6 is H'10, the vector address is H'0420. When the DISEL bit is 0, these bits can be written to.
2	DTVEC2	0	R/W	
1	DTVEC1	0	R/W	
0	DTVEC0	0	R/W	

and interrupt controller priorities have no effect. If there is more than one activation source at the same time, the DTC operates in accordance with the default priorities. Figure 7.2 shows a block diagram of DTC activation source control. For details on the interrupt controller, see the Interrupt Controller.

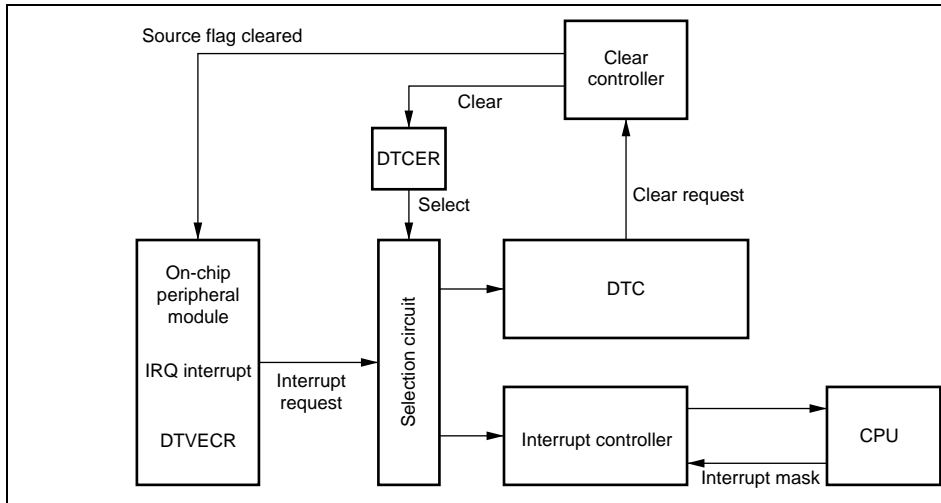


Figure 7.2 Block Diagram of DTC Activation Source Control

vector address corresponding to the interrupt source in the DTC vector table. The DTC then reads the register information from the vector table set for each activation so then reads the register information from that start address.

When the DTC is activated by software, the vector address is obtained from: $H'0400 + (DTVECR[6:0] \times 2)$. For example, if DTVECR is H'10, the vector address is H'0420.

The configuration of the vector address is the same in both normal and advanced mode. The same unit is used in both cases. Specify the lower two bits of the register information start address.

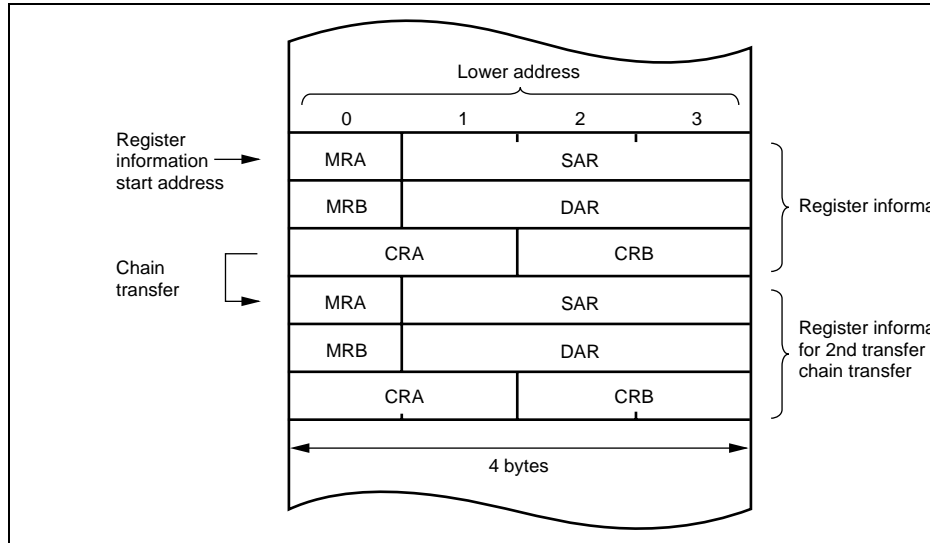


Figure 7.3 DTC Register Information Location in Address Space

	IRQ3	19	H'0426	DTCEA4
A/D converter	ADI	28	H'0438	DTCEA3
FRT	ICIA	48	H'0460	DTCEA2
	ICIB	49	H'0462	DTCEA1
	OCIA	52	H'0468	DTCEA0
	OCIB	53	H'046A	DTCEB7
TMR_0	CMIA0	64	H'0480	DTCEB2
	CMIB0	65	H'0482	DTCEB1
TMR_1	CMIA1	68	H'0488	DTCEB0
	CMIB1	69	H'048A	DTCEC7
TMR_Y	CMIA Y	72	H'0490	DTCEC6
	CMIB Y	73	H'0492	DTCEC5
XBS	IBF1	76	H'0498	DTCEC4
	IBF2	77	H'049A	DTCEC3
SCI_0	RXI0	81	H'04A2	DTCEC2
	TXI0	82	H'04A4	DTCEC1
SCI_1	RXI1	85	H'04AA	DTCEC0
	TXI1	86	H'04AC	DTCED7
SCI_2	RXI2	89	H'04B2	DTCED6
	TXI2	90	H'04B4	DTCED5
IIC_0	IIC I0	92	H'04B8	DTCED4
IIC_1	IIC I1	94	H'04BC	DTCED3
LPC ^{*2}	ERRI	108	H'04D8	DTCEE3
	IBFI1	109	H'04DA	DTCEE2
	IBFI2	110	H'04DC	DTCEE1
	IBFI3	111	H'04DE	DTCEE0

Notes: 1. DTCE bits with no corresponding interrupt are reserved, and only 0 should be written to this bit.

2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

The 24-bit SAR designates the DTC transfer source address, and the 24-bit DAR designates the transfer destination address. After each transfer, SAR and DAR are independently incremented, decremented, or left fixed depending on its register information.

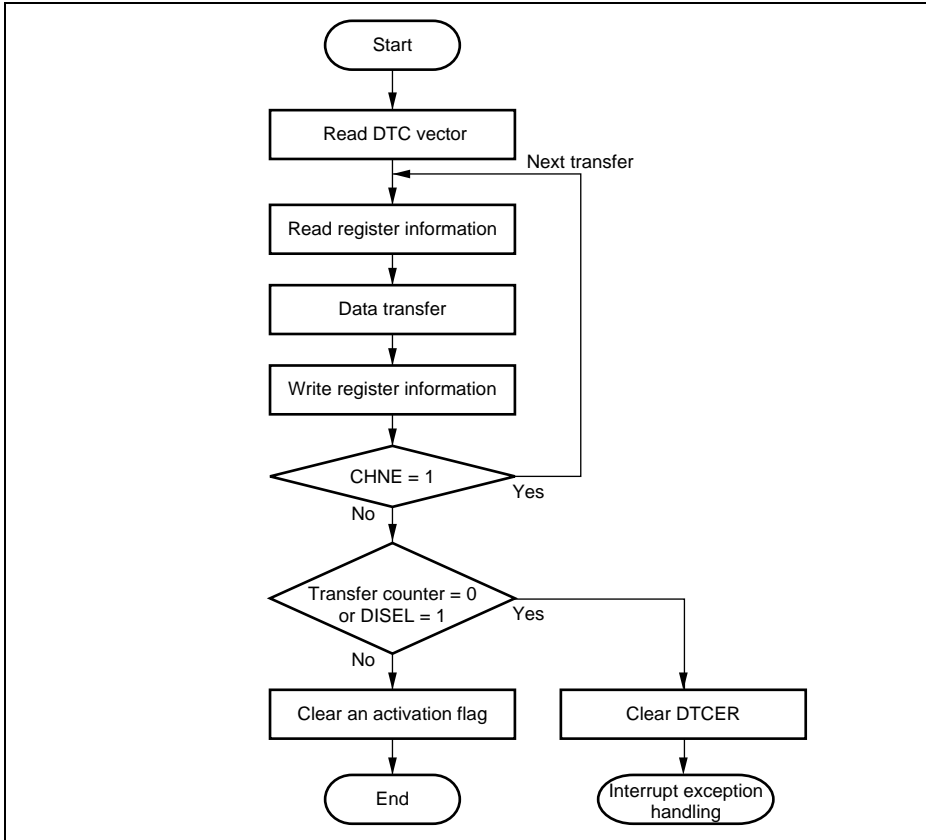


Figure 7.4 DTC Operation Flowchart

DTC source address register	SAR	Transfer source address
DTC destination address register	DAR	Transfer destination address
DTC transfer count register A	CRA	Transfer counter
DTC transfer count register B	CRB	Not used

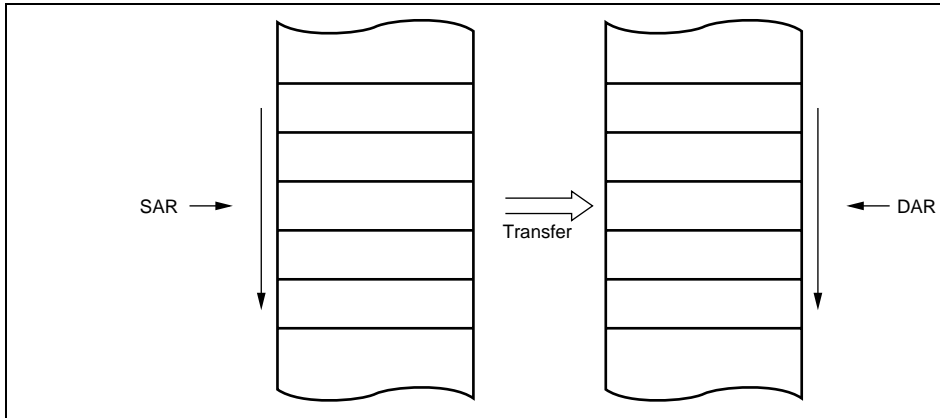


Figure 7.5 Memory Mapping in Normal Mode

Table 7.3 Register Functions in Repeat Mode

Name	Abbreviation	Function
DTC source address register	SAR	Transfer source address
DTC destination address register	DAR	Transfer destination address
DTC transfer count register AH	CRAH	Holds number of transfer
DTC transfer count register AL	CRAL	Transfer Count
DTC transfer count register B	CRB	Not used

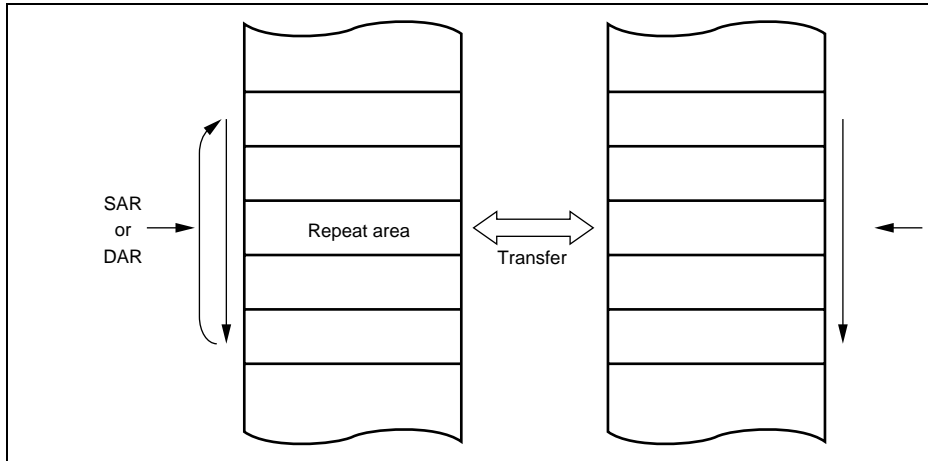


Figure 7.6 Memory Mapping in Repeat Mode

specified number of transfers have been completed, a CPU interrupt is requested.

Table 7.4 Register Functions in Block Transfer Mode

Name	Abbreviation	Function
DTC source address register	SAR	Transfer source address
DTC destination address register	DAR	Transfer destination address
DTC transfer count register AH	CRAH	Holds block size
DTC transfer count register AL	CRAL	Block size counter
DTC transfer count register B	CRB	Transfer counter

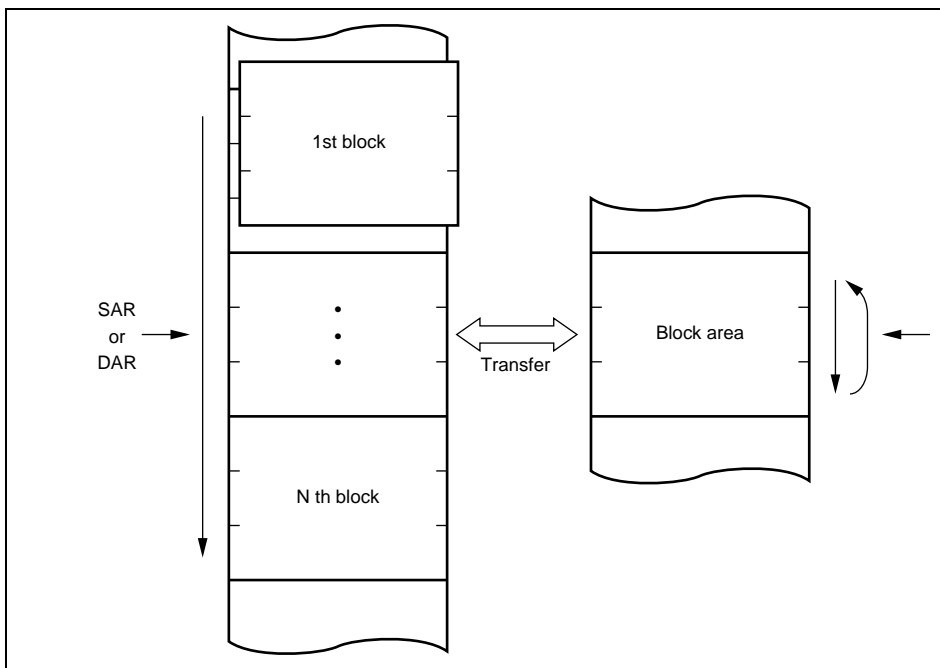


Figure 7.7 Memory Mapping in Block Transfer Mode

been set to 1, DTC reads the next register information located in a consecutive area and continues the data transfer. These sequences are repeated until the CHNE bit is cleared to 0.

In the case of transfer with the CHNE bit set to 1, an interrupt request to the CPU is not generated at the end of the specified number of transfers or by setting of the DISEL bit to 1, and the source flag for the activation source is not affected.

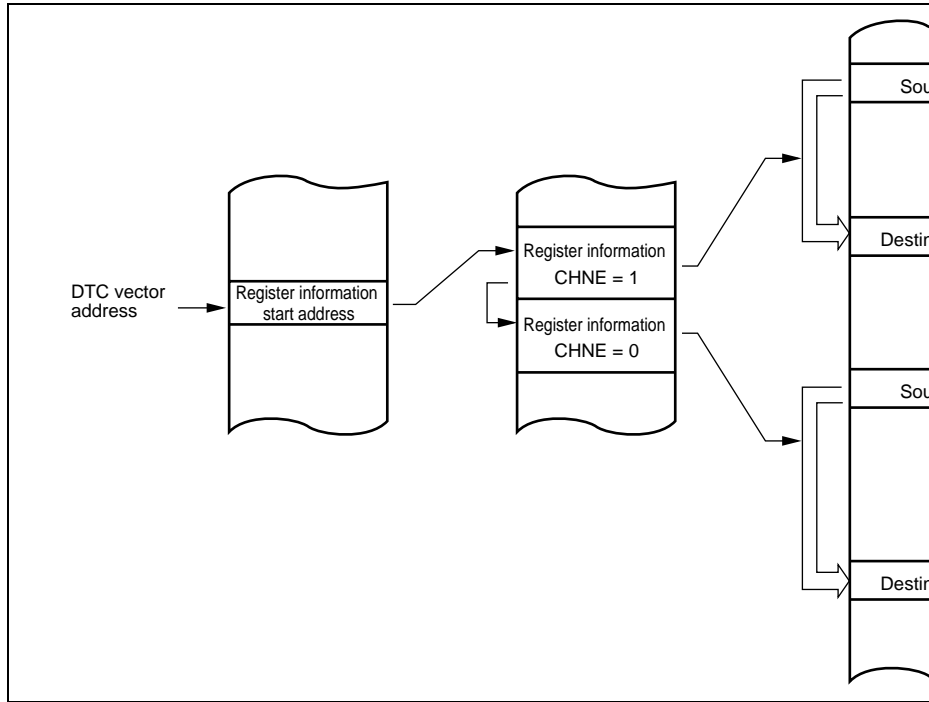


Figure 7.8 Chain Transfer Operation

When the DISEL bit is 1 and one data transfer has been completed, or the specified number of data transfers have been completed, after data transfer ends, the SWDTE bit is held at 1 and a SWDTEND interrupt is generated. The interrupt handling routine will then clear the SWDTE bit to 0.

When the DTC is activated by software, an SWDTEND interrupt is not generated during transfer wait or during data transfer even if the SWDTE bit is set to 1.

7.5.6 Operation Timing

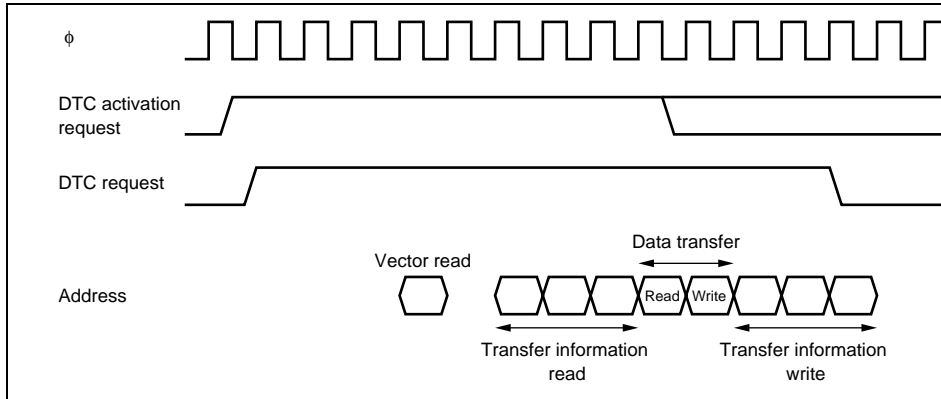


Figure 7.9 DTC Operation Timing (Example in Normal Mode or Repeat Mode)



Figure 7.10 DTC Operation Timing (Example of Block Transfer Mode with Block Size of 2)

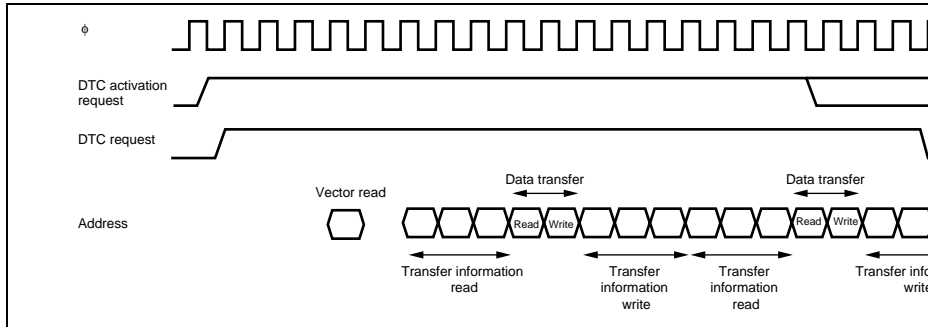


Figure 7.11 DTC Operation Timing (Example of Chain Transfer)

7.5.7 Number of DTC Execution States

Table 7.5 lists the execution status for a single DTC data transfer, and table 7.6 shows of states required for each execution status.

Table 7.6 Number of States Required for Each Execution Status

Object to be Accessed			On-Chip RAM	On-Chip ROM	On-Chip I/O Registers		External Dev		
					8	16	8		
Bus width			32	16	8	16	8		
Access states			1	1	2	2	2	3	2
Execution status	Vector read	S_i	—	1	—	—	4	$6 + 2m$	2
	Register information read/write	S_j	1	—	—	—	—	—	—
	Byte data read	S_k	1	1	2	2	2	$3 + m$	2
	Word data read	S_k	1	1	4	2	4	$6 + 2m$	2
	Byte data write	S_L	1	1	2	2	2	$3 + m$	2
	Word data write	S_L	1	1	4	2	4	$6 + 2m$	2
	Internal operation	S_M	1						

The number of execution states is calculated from using the formula below. Note that Σ is the sum of all transfers activated by one activation source (the number in which the CHNE bit is set plus 1).

$$\text{Number of execution states} = I \cdot S_i + \Sigma (J \cdot S_j + K \cdot S_k + L \cdot S_L) + M \cdot S_M$$

For example, when the DTC vector address table is located in on-chip ROM, normal m and data is transferred from on-chip ROM to an internal I/O register, then the time requ DTC operation is 13 states. The time from activation to the end of data write is 10 states

3. Set the corresponding bit in DTCE to 1.
4. Set the enable bits for the interrupt sources to be used as the activation sources to 1. The interrupt source is activated when an interrupt used as an activation source is generated.
5. After one data transfer has been completed, or after the specified number of data transfers has been completed, the DTCE bit is cleared to 0 and a CPU interrupt is requested. If the DTC is to continue transferring data, set the DTCE bit to 1.

7.6.2 Activation by Software

The procedure for using the DTC with software activation is as follows:

1. Set the MRA, MRB, SAR, DAR, CRA, and CRB register information in on-chip memory.
2. Set the start address of the register information in the DTC vector address.
3. Check that the SWDTE bit is 0.
4. Write 1 to the SWDTE bit and the vector number to DTVECR.
5. Check the vector number written to DTVECR.
6. After one data transfer has been completed, if the DISEL bit is 0 and a CPU interrupt is requested, the SWDTE bit is cleared to 0. If the DTC is to continue transferring data, set the SWDTE bit to 1. When the DISEL bit is 1 or after the specified number of data transfers has been completed, the SWDTE bit is held at 1 and a CPU interrupt is requested.

any value. Set MRB for one data transfer by one interrupt (CHNE = 0, DISEL = 0).
SCI, RDR address in SAR, the start address of the RAM area where the data will be
in DAR, and 128 (H'0080) in CRA. CRB can be set to any value.

2. Set the start address of the register information at the DTC vector address.
3. Set the corresponding bit in DTCER to 1.
4. Set the SCI to the appropriate receive mode. Set the RIE bit in SCR to 1 to enable the
complete (RXI) interrupt. Since the generation of a receive error during the SCI receive
operation will disable subsequent reception, the CPU should be enabled to accept receive
interrupts.
5. Each time the reception of one byte of data has been completed on the SCI, the RDRF
SSR is set to 1, an RXI interrupt is generated, and the DTC is activated. The receive data
is transferred from RDR to RAM by the DTC. DAR is incremented and CRA is decremented.
The RDRF flag is automatically cleared to 0.
6. When CRA becomes 0 after 128 data transfers have been completed, the RDRF flag is
cleared to 0, the DTCE bit is cleared to 0, and an RXI interrupt request is sent to the CPU. The
handling routine will perform wrap-up processing.

- 0). The DTS bit can have any value. Set MRB for one block transfer by one interrupt.
- 0). Set the transfer source address (H'1000) in SAR, the transfer destination address in DAR, and 128 (H'8080) in CRA. Set 1 (H'0001) in CRB.
2. Set the start address of the register information at the DTC vector address (H'04C0).
3. Check that the SWDTE bit in DTVECR is 0. Check that there is currently no transfer by software.
4. Write 1 to the SWDTE bit and the vector number (H'60) to DTVECR. The write data is 0.
5. Read DTVECR again and check that it is set to the vector number (H'60). If it is not, it indicates that the write failed. This is presumably because an interrupt occurred between steps 3 and 4 and led to a different software activation. To activate this transfer, go back to step 4.
6. If the write was successful, the DTC is activated and a block of 128 bytes of data is transferred.
7. After the transfer, an SWDTEND interrupt occurs. The interrupt handling routine sets the SWDTE bit to 0 and perform wrap-up processing.

7.8.2 On-Chip RAM

MRA, MRB, SAR, DAR, CRA, and CRB are all located in on-chip RAM. When the DTC is active, the RAME bit in SYSCR should not be cleared to 0.

7.8.3 DTCE Bit Setting

For DTCE bit setting, use bit manipulation instructions such as BSET and BCLR, for reading and writing. Multiple DTC activation sources can be set at one time (only at the initial setting). After setting, mask all interrupts and write data after executing a dummy read on the relevant register.

7.8.4 Setting Required on Entering Subactive Mode or Watch Mode

Set the MSTP14 bit in MSTPCRH to 1 to make the DTC enter module stop mode, then set the MSTP14 bit to 0. The MSTP14 bit must be set to 1 before making a transition to subactive mode or watch mode.

7.8.5 DTC Activation by Interrupt Sources of SCI, IIC, LPC, or A/D Converter

Interrupt sources of the SCI, IIC, LPC, or A/D converter which activate the DTC are cleared by the DTC reads from or writes to the respective registers, and they cannot be cleared by the DTC reads from or writes to the registers in MRB.

For R3S2100D and R3S2101D.

Table 8.1 is a summary of the port functions. The pins of each port also have other functions. Each port includes a data direction register (DDR) that controls input/output (not provided for input-only port) and data registers (DR, ODR) that store output data.

Ports 1 to 3, 6, A, and B have an on-chip input pull-up MOS function. For ports A and B, the on/off status of the input pull-up MOS is controlled by DDR and ODR. Ports 1 to 3 and 6 have an input pull-up MOS control register (PCR), in addition to DDR, to control the on/off status of the input pull-up MOS.

Ports 1 to 6, 8, 9, A, and B can drive a single TTL load and 30 pF capacitive load. All ports can drive a Darlington transistor when in output mode. Ports 1 to 3 can drive an 80 mA sink current).

Port A input and output use by the VccB power supply, which is independent of the VccA supply. When the VccB voltage is 5V, the pins on port A will be 5-V tolerant.

PA4 to PA7 of port A have bus-buffer drive capability.

P52 in port 5, P97 in port 9, P86 in port 8 and P42 in port 4 are NMOS push-pull outputs. P97, P86 and P42 are thus 5-V tolerant, with DC characteristics that are dependent on the output voltage.

For the P42/SCK2, P52/SCK0, P86/SCK1, and P97 outputs, connect pull-up resistors to raise output-high-level voltage.

		A3	A3/P13/PW3	P13/PW3
		A2	A2/P12/PW2	P12/PW2
		A1	A1/P11/PW1	P11/PW1
		A0	A0/P10/PW0	P10/PW0
Port 2	General I/O port also functioning as address output pin, PWM output pin, and timer connection output pin	A15	A15/P27/PW15/ CBLANK	P27/PW15/CBLANK
		A14	A14/P26/PW14	P26/PW14
		A13	A13/P25/PW13	P25/PW13
		A12	A12/P24/PW12	P24/PW12
		A11	A11/P23/PW11	P23/PW11
		A10	A10/P22/PW10	P22/PW10
		A9	A9/P21/PW9	P21/PW9
		A8	A8/P20/PW8	P20/PW8
Port 3	General I/O port also functioning as data bus input/output, XBS data bus input/output, and LPC input/output pins	D15		P37/HDB7/SERIRQ*
		D14		P36/HDB6/LCLK*
		D13		P35/HDB5/ $\overline{\text{LRESET}}$ *
		D12		P34/HDB4/ $\overline{\text{LFRAME}}$ *
		D11		P33/HDB3/LAD3*
		D10		P32/HDB2/LAD2*
		D9		P31/HDB1/LAD1*
		D8		P30/HDB0/LAD0*

	interface input/output, and IIC_1 input/output pins	P41/TMO0/RxD2/IrRxD P40/TMCI0/TxD2/IrTxD	P43/TMCI1/IrRxQ1/IrTxQ1 HSYNCI P42/TMRI0/SCK2/SDA2 P41/TMO0/RxD2/IrRxD2 P40/TMCI0/TxD2/IrTxD2
Port 5	General I/O port also functioning as SCI_0 input/output and IIC_0 input/output pins	P52/SCK0/SCL0 P51/RxD0 P50/TxD0	
Port 6	General I/O port also functioning as interrupt input, FRT input/output, TMR_X and TMR_Y input/output, timer connection input/output, key-sense interrupt input, and expansion A/D input pins	P67/IRQ7/TMOX/KIN7/CIN7 P66/IRQ6/FTOB/KIN6/CIN6 P65/FTID/KIN5/CIN5 P64/FTIC/KIN4/CIN4/CLAMPO P63/FTIB/KIN3/CIN3/VFBACKI P62/FTIA/TMIY/KIN2/CIN2/VSYNCI P61/FTOA/KIN1/CIN1/VSYNCO P60/FTCI/TMIX/KIN0/CIN0/HFBACKI	
Port 7	General input port also functioning as A/D converter analog input and D/A converter analog output pins	P77/AN7/DA1 P76/AN6/DA0 P75/AN5 P74/AN4 P73/AN3 P72/AN2 P71/AN1 P70/AN0	

		P80	P80/HA0/PME*	
Port 9	General I/O port also functioning as extended data bus control input/output, IIC_0 input/output, subclock input, ϕ output, interrupt input, XBS control input, and A/D converter external trigger input pins	P97/WAIT/SDA0 P96/ ϕ /EXCL $\overline{AS}/\overline{IOS}$ \overline{HWR} \overline{RD} P92/ $\overline{IRQ0}$ P91/ $\overline{IRQ1}$ P90/LWR/ $\overline{IRQ2}/\overline{ADTRG}$	P97/SDA0 P96/ ϕ /EXCL P95/ $\overline{CS1}$ P94/ \overline{IOW} P93/ \overline{IOR} P92/ $\overline{IRQ0}$ P91/ $\overline{IRQ1}$ P90/ $\overline{IRQ2}/\overline{ADTRG}/\overline{ECS2}$	
Port A	General I/O port also functioning as address output, key-sense interrupt input, extended A/D input, and keyboard buffer controller input/output pins	PA7/ $\overline{KIN15}/\overline{CIN15}/\overline{PS2CD}$ PA6/ $\overline{KIN14}/\overline{CIN14}/\overline{PS2CC}$ PA5/ $\overline{KIN13}/\overline{CIN13}/\overline{PS2BD}$ PA4/ $\overline{KIN12}/\overline{CIN12}/\overline{PS2BC}$ PA3/ $\overline{KIN11}/\overline{CIN11}/\overline{PS2AD}$ PA2/ $\overline{KIN10}/\overline{CIN10}/\overline{PS2AC}$ PA1/ $\overline{KIN9}/\overline{CIN9}$ PA0/ $\overline{KIN8}/\overline{CIN8}$	PA7/A23/ $\overline{KIN15}/\overline{CIN15}/\overline{PS2CD}$ PA6/A22/ $\overline{KIN14}/\overline{CIN14}/\overline{PS2CC}$ PA5/A21/ $\overline{KIN13}/\overline{CIN13}/\overline{PS2BD}$ PA4/A20/ $\overline{KIN12}/\overline{CIN12}/\overline{PS2BC}$ PA3/A19/ $\overline{KIN11}/\overline{CIN11}/\overline{PS2AD}$ PA2/A18/ $\overline{KIN10}/\overline{CIN10}/\overline{PS2AC}$ PA1/A17/ $\overline{KIN9}/\overline{CIN9}$ PA0/A16/ $\overline{KIN8}/\overline{CIN8}$	PA7/ $\overline{KIN15}/\overline{CIN15}/\overline{PS2CD}$ PA6/ $\overline{KIN14}/\overline{CIN14}/\overline{PS2CC}$ PA5/ $\overline{KIN13}/\overline{CIN13}/\overline{PS2BD}$ PA4/ $\overline{KIN12}/\overline{CIN12}/\overline{PS2BC}$ PA3/ $\overline{KIN11}/\overline{CIN11}/\overline{PS2AD}$ PA2/ $\overline{KIN10}/\overline{CIN10}/\overline{PS2AC}$ PA1/ $\overline{KIN9}/\overline{CIN9}$ PA0/ $\overline{KIN8}/\overline{CIN8}$

		$\overline{\text{PB1/D1/WUE1}}$ * $\overline{\text{PB0/D0/WUE0}}$ *	$\overline{\text{PB1/WUE1}}$ */HIRQ4/ LSCI* $\overline{\text{PB0/WUE0}}$ */HIRQ3/ LSMI*
--	--	--	--

Note: * Not supported by the H8S/2148B and H8S/2145B (5-V version).

- Port 1 pull-up MOS control register (P1PCR)

8.2.1 Port 1 Data Direction Register (P1DDR)

P1DDR specifies input or output for the pins of port 1 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	P17DDR	0	W	In mode 1:
6	P16DDR	0	W	Each pin of port 1 is address output reg
5	P15DDR	0	W	the set value of P1DDR.
4	P14DDR	0	W	In modes 2 and 3 (EXPE=1):
3	P13DDR	0	W	The corresponding port 1 pins are addre
2	P12DDR	0	W	or PWM output ports when P1DDR bits
1	P11DDR	0	W	1, and input ports when cleared to 0.
0	P10DDR	0	W	In modes 2 and 3 (EXPE=0):
				The corresponding port 1 pins are outpu
				PWM outputs when the P1DDR bits are
				and input ports when cleared to 0.

4	P14DR	0	R/W
3	P13DR	0	R/W
2	P12DR	0	R/W
1	P11DR	0	R/W
0	P10DR	0	R/W

8.2.3 Port 1 Pull-Up MOS Control Register (P1PCR)

P1PCR controls the on/off status of the port 1 on-chip input pull-up MOSs.

Bit	Bit Name	Initial Value	R/W	Description
7	P17PCR	0	R/W	When the pins are in input state, the corresponding input pull-up MOS is turned on when a P1PCR bit is set to 1.
6	P16PCR	0	R/W	
5	P15PCR	0	R/W	
4	P14PCR	0	R/W	
3	P13PCR	0	R/W	
2	P12PCR	0	R/W	
1	P11PCR	0	R/W	
0	P10PCR	0	R/W	

P1MDDR	—	—	0	1	—	0
OEn	—	—	0	1	—	0
Pin Function	A7 to A0 output pins	P17 to P10 input pins	A7 to A0 output pins	PW7 to PW0 output pins	P17 to P10 input pins	P17 to P10 output pins

Note: n = 7 to 0

8.2.5 Port 1 Input Pull-Up MOS

Port 1 has an on-chip input pull-up MOS function that can be controlled by software. The pull-up MOS function can be specified as on or off on a bit-by-bit basis.

Table 8.2 summarizes the input pull-up MOS states.

Table 8.2 Input Pull-Up MOS States (Port 1)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Other Operation
1	Off	Off	Off	Off
2, 3			On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, P1DDR = 0, and P1PCR = 1; otherwise off.

- Port 2 data register (P2DDR)
- Port 2 pull-up MOS control register (P2PCR)

8.3.1 Port 2 Data Direction Register (P2DDR)

P2DDR specifies input or output for the pins of port 2 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	P27DDR	0	W	In Mode 1:
6	P26DDR	0	W	The corresponding port 2 pins are address outputs, regardless of the P2DDR settings.
5	P25DDR	0	W	Modes 2 and 3 (EXPE = 1):
4	P24DDR	0	W	The corresponding port 2 pins are address outputs or PWM outputs when P2DDR is set to 1, and input ports when cleared to 0.
3	P23DDR	0	W	P24 are switched from address output ports by setting the IOSE bit to 1.
2	P22DDR	0	W	P27 can be used as an on-chip peripheral output pin regardless of the P27DDR settings to ensure normal access to external signals should not be set as an on-chip peripheral module output pin when port 2 pins are address output pins.
1	P21DDR	0	W	Modes 2 and 3 (EXPE = 0):
0	P20DDR	0	W	The corresponding port 2 pins are output ports or PWM outputs when P2DDR bits are set to 1, and input ports when cleared to 0. P27 can be used as an on-chip peripheral output pin regardless of the P27DDR settings.

4	P24DR	0	R/W	the pin states are read.
3	P23DR	0	R/W	
2	P22DR	0	R/W	
1	P21DR	0	R/W	
0	P20DR	0	R/W	

8.3.3 Port 2 Pull-Up MOS Control Register (P2PCR)

P2PCR controls the port 2 on-chip input pull-up MOSs.

Bit	Bit Name	Initial Value	R/W	Description
7	P27PCR	0	R/W	In modes 2 and 3, the input pull-up MOS is turned on when a P2PCR bit is set to 1 in the state.
6	P26PCR	0	R/W	
5	P25PCR	0	R/W	
4	P24PCR	0	R/W	
3	P23PCR	0	R/W	
2	P22PCR	0	R/W	
1	P21PCR	0	R/W	
0	P20PCR	0	R/W	

the P27DDR bit, and operating mode.

Operating Mode	Mode 1	Mode 2, 3 (EXPE = 1)					Mode 2, 3 (EXP		
CBOE	—	0				1	0		
P27DDR	—	0	1			—	0	1	
OE15	—	—	0		1	—	—	0	1
IOSE	—	—	0	1	—	—	—	—	—
Pin Function	A15 output pin	P27 input pin	A15 output pin	P27 output pin	PW15 output pin	CBLANK output pin	P27 input pin	P27 output pin	PW output pin

- P26/A14/PW14, P25/A13/PW13, P24/A12/PW12

The pin function is switched as shown below according to the combination of the P2nDDR bit in SYSCR, the OEm bit in PWOERB of PWM, the P2nDDR bit, and operating mode.

Operating Mode	Mode 1	Mode 2, 3 (EXPE = 1)				Mode 2, 3 (E	
P2nDDR	—	0	1			0	1
OEm	—	—	0		1	—	0
IOSE	—	—	0	1	—	—	1
Pin Function	A14 to A12 output pins	P26 to P24 input pins	A14 to A12 output pins	P26 to P24 output pins	PW14 to PW12 output pins	P26 to P24 input pins	P26 to P24 output pins

Note: n = 6 to 4
m = 14 to 12

Pin Function	A11 to A8 output pins	P23 to P20 input pins	A11 to A8 output pins	PW11 to PW8 output pins	P23 to P20 input pins	P23 to P20 output pins
--------------	-----------------------------	-----------------------------	-----------------------------	----------------------------------	-----------------------------	---------------------------------

Note: n = 3 to 0
m = 11 to 8

8.3.5 Port 2 Input Pull-Up MOS

Port 2 has an on-chip input pull-up MOS function that can be controlled by software. The pull-up MOS function can be specified as on or off on a bit-by-bit basis.

Table 8.3 summarizes the input pull-up MOS states.

Table 8.3 Input Pull-Up MOS States (Port 2)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Other Operation
1	Off	Off	Off	Off
2, 3			On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, P2DDR = 0, and P2PCR = 1; otherwise off.

- Port 3 pull-up MOS control register (P3PCR)

8.4.1 Port 3 Data Direction Register (P3DDR)

P3DDR specifies input or output for the pins of port 3 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	P37DDR	0	W	Modes 1, 2, and 3 (EXPE = 1)
6	P36DDR	0	W	The input/output direction specified by
5	P35DDR	0	W	ignored, and pins automatically function
4	P34DDR	0	W	I/O pins.
3	P33DDR	0	W	Modes 2 and 3 (EXPE = 0)
2	P32DDR	0	W	The corresponding port 3 pins are output
1	P31DDR	0	W	when P3DDR bits are set to 1, and input
0	P30DDR	0	W	when cleared to 0.

4	P34DR	0	R/W	the pin states are read.
3	P33DR	0	R/W	
2	P32DR	0	R/W	
1	P31DR	0	R/W	
0	P30DR	0	R/W	

8.4.3 Port 3 Pull-Up MOS Control Register (P3PCR)

P3PCR controls the port 3 on-chip input pull-up MOSs on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	P37PCR	0	R/W	In modes 2 and 3 (when EXPE = 0), the pull-up MOS is turned on when a P3PCR bit is set to 1 in the input port state.
6	P36PCR	0	R/W	
5	P35PCR	0	R/W	The input pull-up MOS function cannot be used when the host interface is enabled.
4	P34PCR	0	R/W	
3	P33PCR	0	R/W	
2	P32PCR	0	R/W	
1	P31PCR	0	R/W	
0	P30PCR	0	R/W	

Note: * Not supported by the H8S/2148B and H8S/2145B (5-V version).

Operating Mode	Mode 1, 2, 3 (EXPE = 1)	Mode 2, 3 (EXPE = 0)			
LPCmE	All 0	All 0			
HI12E	0	0		1	
P3nDDR	—	0	1	—	
Pin Function	D15 to D8 input/output pins	P37 to P30 input pins	P37 to P30 output pins	HDB7 to HDB0 input/output pins	in

Notes: The combination of bits not described in the above table must not be used.
 m = 3 to 1: LPC input/output pins (SERIRQ, LCLK, $\overline{\text{LRESET}}$, $\overline{\text{LFRAME}}$, LAD3) when at least one of LPC3E to LPC1E is set to 1.
 n = 7 to 0

8.4.5 Port 3 Input Pull-Up MOS

Port 3 has an on-chip input pull-up MOS function that can be controlled by software. The pull-up MOS function can be specified as on or off on a bit-by-bit basis.

Table 8.4 summarizes the input pull-up MOS states.

Table 8.4 Input Pull-Up MOS States (Port 3)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Other Operat
1, 2, 3 (EXPE = 1)	Off	Off	Off	Off
2, 3 (EXPE = 0)			On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, P3DDR = 0, and P3PCR = 1; otherwise off.

- Port 4 data register (P4DR)

8.5.1 Port 4 Data Direction Register (P4DDR)

P4DDR specifies input or output for the pins of port 4 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	P47DDR	0	W	When a bit in P4DDR is set to 1, the corresponding pin functions as an output when cleared to 0, as an input port.
6	P46DDR	0	W	
5	P45DDR	0	W	As 14-bit PWM and SCI_2 are initialized in software standby mode, the pin states are determined by the TMR_0, TMR_1, XBR, P4DDR, and P4DR specifications.
4	P44DDR	0	W	
3	P43DDR	0	W	
2	P42DDR	0	W	
1	P41DDR	0	W	
0	P40DDR	0	W	

4	P44DR	0	R/W
3	P43DR	0	R/W
2	P42DR	0	R/W
1	P41DR	0	R/W
0	P40DR	0	R/W

the pin states are read.

8.5.3 Pin Functions

- P47/PWX1

The pin function is switched as shown below according to the combination of the DACR of the 14-bit PWM and the P47DDR bit.

OEB	0		1
P47DDR	0	1	—
Pin Function	P47 input pin	P47 output pin	PWX1 output

- P46/PWX0

The pin function is switched as shown below according to the combination of the DACR of the 14-bit PWM and the P46DDR bit.

OEA	0		1
P46DDR	0	1	—
Pin Function	P46 input pin	P46 output pin	PWX0 output

Note: * When bits CCLR1 and CCLR0 in TCR1 of TMR_1 are set to 1, this pin is used as the TMR11 input pin. It can also be used as the CSYNCl input pin.

- P44/TMO1/HIRQ1/HSYNCO

The pin function is switched as shown below according to the combination of the HSYSCR2, the OS3 to OS0 bits in TCSR of TMR_1, the HOE bit in TCONRO of the connection function, and the P44DDR bit.

HOE	0				
OS3 to OS0	All 0			Not all 0	
P44DDR	0	1		—	
HI12E	—	0	1	—	
Pin Function	P44 input pin	P44 output pin	HIRQ1 output pin	TMO1 output pin	H c

- P43/TMC11/HIRQ11/HSYNCl

The pin function is switched as shown below according to the combination of the HSYSCR2 and the P43DDR bit.

P43DDR	0	1	
HI12E	—	0	1
Pin Function	P43 input pin	P43 output pin	HIRQ11 ou
	TMC11 input pin, HSYNCl input pin*		

Note: * When the external clock is selected by bits CKS2 to CKS0 in TCR1 of TMR_1, this pin is used as the TMC11 input pin. It can also be used as the HSYNCl input pin.

CKE0	0		1	—	—
P42DDR	0	1	—	—	—
Pin Function	P42 input pin	P42 output pin	SCK2 output pin	SCK2 output pin	SCK2 input pin
	TMR10 input pin*				

Note: * When this pin is used as the SDA1 I/O pin, bits CKE1 and CKE0 in SCR of bit C/ \bar{A} in SMR of SCI_2 must all be cleared to 0. SDA1 is an NMOS-only and has direct bus drive capability.

When bits CCLR1 and CCLR0 in TCR0 of TMR_0 are set to 1, this pin is used as TMR10 input pin.

When the P42 output pin and SCK2 output pin are set, the output type is N-channel open-drain pull output.

- P41/TMO0/RxD2/IrRxD

The pin function is switched as shown below according to the combination of the RE bits in TCSR of TMR0, the RE bit in SCR of SCI_2 and the P41DDR bit.

OS3 to OS0	All 0			M
RE	0		1	
P41DDR	0	1	—	
Pin Function	P41 input pin	P41 output pin	RxD2/IrRxD input pin	o

Note: When this pin is used as the TMO0 output pin, bit RE in SCR of SCI_2 must be cleared to 0.

Note: * When an external clock is selected with bits CKS2 to CKS0 in TCR0 of TMR is used as the TMC10 input pin.

8.6 Port 5

Port 5 is a 3-bit I/O port. Port 5 pins also function as SCI_0 I/O pins, and the IIC_0 I/O and SCK0 are NMOS push-pull outputs, and SCL0 is an NMOS open-drain output. Port 5 has the following registers.

- Port 5 data direction register (P5DDR)
- Port 5 data register (P5DR)

8.6.1 Port 5 Data Direction Register (P5DDR)

P5DDR specifies input or output for the pins of port 5 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 1	—	Reserved The initial value must not be changed.
2	P52DDR	0	W	The corresponding port 5 pins are output when P5DDR bits are set to 1, and input when cleared to 0. As SCI_0 is initialized in software standby mode, the pin states are determined by the IIC_0 ICCR, P5DDR and P5DR specifications.
1	P51DDR	0	W	
0	P50DDR	0	W	

1	P51DR	0	R/W
0	P50DR	0	R/W

set to 1, the P5DR values are read directly regardless of the actual pin states. If a read operation is performed while P5DDR bits are cleared, the pin states are read.

8.6.3 Pin Functions

- P52/SCK0/SCL0

The pin function is switched as shown below according to the combination of the CKE0 bits in SCR of SCI_0, the C/A bit in SMR of SCI_0, the ICE bit in ICCR of SCI_0, and the P52DDR bit.

ICE	0				
CKE1	0				1
C/A	0			1	—
CKE0	0		1	—	—
P52DDR	0	1	—	—	—
Pin Function	P52 input pin	P52 output pin	SCK0 output pin	SCK0 output pin	SCK0 input pin

Note: When this pin is used as the SCL0 I/O pin, bits CKE1 and CKE0 in SCR of SCI_0 and C/A bit in SMR of SCI_0 must all be cleared to 0.

SCL0 is an NMOS open-drain output, and has direct bus drive capability.

When set as the P52 output pin or SCK0 output pin, this pin is an NMOS push-pull output.

- P50/TxD0

The pin function is switched as shown below according to the combination of the TSCR of SCI_0 and the P50DDR bit.

TE	0		1
P50DDR	0	1	—
Pin Function	P50 input pin	P50 output pin	TxD0 output

8.7 Port 6

Port 6 is an 8-bit I/O port. Port 6 pins also function as the FRT I/O pins, TMR_X I/O pins, TMR_Y input pin, timer connection I/O pins, key-sense interrupt input pins, expansion converter input pins, and external interrupt input pins. The port 6 input level can be switched through four stages. Port 6 pin functions are the same in all operating modes. For details on the control register 2 (SYSCR2), refer to section 18, Host Interface X-Bus Interface (XBS) and the following registers.

- Port 6 data direction register (P6DDR)
- Port 6 data register (P6DR)
- Port 6 pull-up MOS control register (KMPCR)
- System control register 2 (SYSCR2)

4	P64DDR	0	W
3	P63DDR	0	W
2	P62DDR	0	W
1	P61DDR	0	W
0	P60DDR	0	W

8.7.2 Port 6 Data Register (P6DR)

P6DR stores output data for port 6.

Bit	Bit Name	Initial Value	R/W	Description
7	P67DR	0	R/W	If a port 6 read is performed while P6DDR bits are set to 1, the P6DR values are read directly regardless of the actual pin states. If a read is performed while P6DDR bits are cleared, the pin states are read.
6	P66DR	0	R/W	
5	P65DR	0	R/W	
4	P64DR	0	R/W	
3	P63DR	0	R/W	
2	P62DR	0	R/W	
1	P61DR	0	R/W	
0	P60DR	0	R/W	

4	KM4PCR	0	R/W
3	KM3PCR	0	R/W
2	KM2PCR	0	R/W
1	KM1PCR	0	R/W
0	KM0PCR	0	R/W

8.7.4 Pin Functions

- P67/TMOX/CIN7/ $\overline{\text{KIN7}}$ / $\overline{\text{IRQ7}}$

The pin function is switched as shown below according to the combination of the OS3 to OS0 bits in TCSR of TMR_X and the P67DDR bit.

OS3 to OS0	All 0		Not a
P67DDR	0	1	—
Pin Function	P67 input pin	P67 output pin	TMOX ou
	$\overline{\text{IRQ7}}$ input pin, $\overline{\text{KIN7}}$ input pin, CIN7 input pin*		

Note: * This pin is used as the $\overline{\text{IRQ7}}$ input pin when bit IRQ7E is set to 1 in IER. It can also be used as the $\overline{\text{KIN7}}$ or CIN7 input pin.

Note: * This pin is used as the $\overline{\text{IRQ6}}$ input pin when bit IRQ6E is set to 1 in IER when the KMIMR6 bit in KMIMR is 0. It can always be used as the $\overline{\text{KIN6}}$ or CIN6 input pin.

- P65/FTID/CIN5/ $\overline{\text{KIN5}}$

P65DDR	0	1
Pin Function	P65 input pin	P65 output pin
	FTID input pin, $\overline{\text{KIN5}}$ input pin, CIN5 input pin*	

Note: * This pin can always be used as the FTID, $\overline{\text{KIN5}}$, or CIN5 input pin.

- P64/FTIC/CIN4/ $\overline{\text{KIN4}}$ /CLAMPO

The pin function is switched as shown below according to the combination of the CLOE and TCONRO of the timer connection function and the P64DDR bit.

CLOE	0		1
P64DDR	0	1	—
Pin Function	P64 input pin	P64 output pin	CLAMPO
	FTIC input pin, $\overline{\text{KIN4}}$ input pin, CIN4 input pin*		

Note: * This pin can always be used as the FTIC, $\overline{\text{KIN4}}$, or CIN4 input pin.

- P63/FTIB/CIN3/ $\overline{\text{KIN3}}$ /VFBACKI

P63DDR	0	1
Pin Function	P63 input pin	P63 output pin
	FTIB input pin, VFBACKI input pin, $\overline{\text{KIN3}}$ input pin, CIN3 input pin	

Note: * This pin can always be used as the FTIB, $\overline{\text{KIN3}}$, CIN3, or VFBACKI input pin.

- P61/FTOA/CIN1/KIN1/VSYNCO

The pin function is switched as shown below according to the combination of the OEA bit in TOCR of the FRT, the VOE bit in TCONRO of the timer connection function, and P61DDR bit.

VOE	0			
OEA	0		1	
P61DDR	0	1	—	
Pin Function	P61 input pin	P61 output pin	FTOA output pin	VSYNCO
	$\overline{\text{KIN1}}$ input pin, CIN1 input pin*			

Note: * When this pin is used as the VSYNCO pin, bit OEA in TOCR of the FRT must be cleared to 0. This pin can always be used as the $\overline{\text{KIN1}}$ or CIN1 input pin.

- P60/FTCI/CIN0/ $\overline{\text{KIN0}}$ /HFBACKI/TMIX

P60DDR	0	1
Pin Function	P60 input pin	P60 output pin
	FTCI input pin, HFBACKI input pin, TMIX input pin, $\overline{\text{KIN0}}$ input pin, CIN0 input pin*	

Note: * This pin is used as the FTCL input pin when an external clock is selected with CKS0 in TCR of the FRT. It can always be used as the TMIX, $\overline{\text{KIN0}}$, CIN0, or HFBACKI input pin.

Table 8.5 Input Pull-Up MOS States (Port 6)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Other Operati
1 to 3	Off	Off	On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, P6DDR = 0, and KMPCR = 1; otherwise

8.8 Port 7

Port 7 is an 8-bit input only port. Port 7 pins also function as the A/D converter analog and D/A converter analog output pins. Port 7 functions are the same in all operating modes. Port 7 has the following register.

- Port 7 input data register (P7PIN)

4	P74PIN	Undefined*	R	changed.
3	P73PIN	Undefined*	R	
2	P72PIN	Undefined*	R	
1	P71PIN	Undefined*	R	
0	P70PIN	Undefined*	R	

Note: * Determined by the pin states of P77 to P70.

8.8.2 Pin Functions

- P77/AN7/DA1

The pin function is switched as shown below according to the combination of the DACR of the D/A converter and the DAOE1 bit.

DAOE1	0		1
DAE	0	1	—
Pin Function	P77 input pin	DA1 input pin	DA1 output pin
	AN7 input pin*		

Note: * This pin can always be used as the AN7 input pin.

- P76/AN6/DA0

The pin function is switched as shown below according to the combination of the DACR of the D/A converter and the DAOE0 bit.

DAOE0	0		1
DAE	0	1	—
Pin Function	P76 input pin	DA0 output pin	DA0 output pin
	AN6 input pin*		

Note: * This pin can always be used as the AN6 input pin.

Port 8 is an 8-bit I/O port. Port 8 pins also function as SCI_1 I/O pins, the IIC_1 I/O pins, LPC I/O pins, and interrupt input pins. The output type of P86 and SCK1 is NMOS open drain output. The output type of SCL1 is NMOS open drain output and direct bus driving is not supported. Port 8 pin functions are the same in all operating modes except host interface function. The following registers control the port 8 pins.

- Port 8 data direction register (P8DDR)
- Port 8 data register (P8DR)

8.9.1 Port 8 Data Direction Register (P8DDR)

P8DDR specifies input or output for the pins of port 8 on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved The initial value must not be changed.
6	P86DDR	0	W	P8DDR has the same address as PBI. When read, the port B state will be returned.
5	P85DDR	0	W	The corresponding port 8 pins are output when P8DDR bits are set to 1, and input when cleared to 0.
4	P84DDR	0	W	
3	P83DDR	0	W	
2	P82DDR	0	W	
1	P81DDR	0	W	
0	P80DDR	0	W	

5	P85DR	0	R/W
4	P84DR	0	R/W
3	P83DR	0	R/W
2	P82DR	0	R/W
1	P81DR	0	R/W
0	P80DR	0	R/W

set to 1, the P8DR values are read directly regardless of the actual pin states. If a read operation is performed while P8DDR bits are cleared, the pin states are read.

8.9.3 Pin Functions

- P86/ $\overline{\text{IRQ5}}$ / SCK1/SCL1

The pin function is switched as shown below according to the combination of the CKE0 bits in SCR of SCI_1, the C/ $\overline{\text{A}}$ bit in SMR of SCI_1, the ICE bit in ICCR of ICCR, and the P86DDR bit.

ICE	0				
CKE1	0			1	
C/ $\overline{\text{A}}$	0		1	—	
CKE0	0		1	—	—
P86DDR	0	1	—	—	—
Pin Function	P86 input pin	P86 output pin	SCK1 output pin	SCK1 output pin	SCK1 input pin
	$\overline{\text{IRQ5}}$ input pin*				

Note: * When the IRQ5E bit in IER is set to 1, this pin is used as the $\overline{\text{IRQ5}}$ input pin. When the P86 output pin and SCK1 are set, the output type is NMOS push-pull output. SCL1 is an NMOS-only I/O pin and has direct bus drive capability.

Note: * When the IRQ4E bit in IER is set to 1, this pin is used as the $\overline{\text{IRQ4}}$ input pin

- P84/ $\overline{\text{IRQ3}}$ /TxD1

The pin function is switched as shown below according to the combination of the P84DDR bit of SCI_1 and the P84DDR bit.

TE	0		1
P84DDR	0	1	—
Pin Function	P84 input pin	P84 output pin	TxD1 output pin
	$\overline{\text{IRQ3}}$ input pin*		

Note: * When the IRQ3E bit in IER is set to 1, this pin is used as the $\overline{\text{IRQ3}}$ input pin

- P83/ $\overline{\text{LPCPD}}$ *²

The pin function is switched as shown below according to the P83DDR bit.

P83DDR	0	1
Pin Function	P83 input pin	P83 output pin
	$\overline{\text{LPCPD}}$ input pin* ^{1*2}	

Notes: 1. When at least one of bits LPC3E to LPC1E is set to 1 in HICR0, this pin is used as the $\overline{\text{LPCPD}}$ input pin. The $\overline{\text{LPCPD}}$ input pin can only be used in mode 2 or 3 (EUSART).

2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

P82DDR	0	1	0	1	—
Pin Function	P82 input pin	P82 output pin	P82 input pin	P82 output pin	HIFSD input pin

Notes: The HIFSD input pin and CLKRUN I/O pin can only be used in mode 2 or 3 (EX).

1. When at least one of bits LPC3E to LPC1E is set to 1, bits HI12E and P82DDR should be cleared to 0.
2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

- P81/ $\overline{\text{CS2}}$ /GA20

The pin function is switched as shown below according to the combination of the HI12E, SYSCR2, the CS2E bit in SYSCR, the FGA20E bit in HICR, the FGA20E bit in HICR, and the P81DDR bit.

FGA20E (LPC)	0						
HI12E	0		1				
FGA20E (XBS)	—		0			1	
CS2E	—		0		1	—	
P81DDR	0	1	0	1	—	0	1
Pin Function	P81 input pin	P81 output pin	P81 input pin	P81 output pin	$\overline{\text{CS2}}$ input pin ^{*2}	P81 input pin	GA20 output pin
	GA20 input pin ^{*2}						

Notes: 1. When bit FGA20E is set to 1 in HICR0, bits HI12E and P81DDR should be 0.

2. The GA20 output pin and $\overline{\text{CS2}}$ input pin can only be used in mode 2 or 3 (EX).

- Notes:
1. When bit PMEE is set to 1 in HICR0, bits HI12E and P80DDR should be cleared.
 2. The HA0 input pin can only be used in mode 2 or 3 (EXPE = 0).
 3. Not supported by the H8S/2148B and H8S/2145B (5-V version).

8.10 Port 9

Port 9 is an 8-bit I/O port. Port 9 pins also function as external interrupt input pins, the A/D converter input pin, host interface (XBS) input pins, the IIC_0 I/O pin, the subclock input pin, control signal I/O pins, and the system clock (ϕ) output pin. P97 is an NMOS push-pull output, SDA0 is an NMOS open-drain output, and has direct bus drive capability. Port 9 has two registers.

- Port 9 data direction register (P9DDR)
- Port 9 data register (P9DR)

5	P95DDR	0	W	Pin P97 functions as a bus control input ($\overline{\text{WAIT}}$), IIC_0 I/O pin (SDA0), or an I/O port, according to mode setting. When P97 functions as an I/O port, it becomes an output port when P97DDR is set to 1, and an input port when P97DDR is cleared to 0.
4	P94DDR	0	W	Pin P96 functions as the ϕ output pin when P96DDR is set to 1, and as the subclock input (EXCL) or an I/O port when P96DDR is cleared to 0.
3	P93DDR	0	W	
2	P92DDR	0	W	Pin P95 functions as the ϕ output pin when P95DDR is set to 1, and as the subclock input (EXCL) or an I/O port when P95DDR is cleared to 0.
1	P91DDR	0	W	
0	P90DDR	0	W	Pin P96 functions as the ϕ output pin when P96DDR is set to 1, and as the subclock input (EXCL) or an I/O port when P96DDR is cleared to 0.

Pins P95 to P93 automatically become bus control outputs ($\overline{\text{AS}}/\overline{\text{IOS}}$, $\overline{\text{HWR}}$, $\overline{\text{RD}}$), regardless of the input/output direction indicated by P95DDR to P93DDR.

Pins P92 and P91 become output ports when P92DDR and P91DDR are set to 1, and input ports when P92DDR and P91DDR are cleared to 0.

When the ABW bit in WSCR is cleared to 0, pin P90 becomes a bus control output ($\overline{\text{LWR}}$), regardless of the input/output direction indicated by P90DDR. When the ABW bit is 1, pin P90 becomes an output port when P90DDR is set to 1, and an input port if P90DDR is cleared to 0.

Modes 2 and 3 (EXPE = 0):

When the corresponding P9DDR bits are set to 1, pin P96 functions as the ϕ output pin and pins P97 to P99 become output ports. When P9DDR bits are cleared to 0, the corresponding pins become input ports.

Note: * The initial value of P96DDR is 1 (mode 1) or 0 (modes 2 and 3).

4	P94DR	0	R/W	while P9DDR bits are cleared to 0, the
3	P93DR	0	R/W	are read.
2	P92DR	0	R/W	For P96, the pin state is always read.
1	P91DR	0	R/W	
0	P90DR	0	R/W	

Note: * The initial value of bit 6 is determined according to the P96 pin state.

8.10.3 Pin Functions

- P97/ $\overline{\text{WAIT}}$ /SDA0

The pin function is switched as shown below according to the combination of operating mode, the WMS1 bit in WSCR, the ICE bit in ICCR of IIC_0, and the P97DDR bit.

Operating Mode	Modes 1, 2, 3 (EXPE = 1)				Modes 2, 3 (EXPE = 0)	
	0		1		—	
WMS1	0		1		—	
ICE	0		1		0	
P97DDR	0	1	—	—	0	1
Pin Function	P97 input pin	P97 output pin	SDA0 I/O pin	$\overline{\text{WAIT}}$ input pin	P97 input pin	P97 output pin

Note: When this pin is set as the P97 output pin, it is an NMOS push-pull output. SDA0 is an NMOS open-drain output, and has direct bus drive capability.

- P95/ \overline{AS} / \overline{IOS} / $\overline{CS1}$

The pin function is switched as shown below according to the combination of operation mode, the IOSE bit in SYSCR, the HI12E bit in SYSCR2, and the P95DDR bit.

Operating Mode	Modes 1, 2, 3 (EXPE = 1)		Modes 2, 3 (EXPE = 0)		
	HI12E	—		0	
P95DDR	—		0	1	
IOSE	0	1	—	—	
Pin Function	\overline{AS} output pin	\overline{IOS} output pin	P95 input pin	P95 output pin	i

- P94/ \overline{HWR} / \overline{IOW}

The pin function is switched as shown below according to the combination of operation mode, the HI12E bit in SYSCR2, and the P94DDR bit.

Operating Mode	Modes 1, 2, 3 (EXPE = 1)		Modes 2, 3 (EXPE = 0)		
	HI12E	—		0	
P94DDR	—		0	1	
Pin Function	\overline{HWR} output pin		P94 input pin	P94 output pin	i

Pin Function	RD output pin	P93 input pin	P93 output pin	TC
--------------	---------------	---------------	----------------	----

- P92/ $\overline{\text{IRQ0}}$

P92DDR	0	1
Pin Function	P92 input pin	P92 output pin
	$\overline{\text{IRQ0}}$ input pin*	

Note: * When bit IRQ0E in IER is set to 1, this pin is used as the $\overline{\text{IRQ0}}$ input pin.

- P91/ $\overline{\text{IRQ1}}$

P91DDR	0	1
Pin Function	P91 input pin	P91 output pin
	$\overline{\text{IRQ1}}$ input pin*	

Note: * When bit IRQ1E in IER is set to 1, this pin is used as the $\overline{\text{IRQ1}}$ input pin.



HI12E	—					Any one 0
FGA20E	—					
CS2E	—					
P90DDR	—	0	1	0	1	
Pin Function	LWR output pin	P90 input pin	P90 output pin	P90 input pin	P90 output pin	
		IRQ2 input pin, ADTRG input pin*				

Note: * When the IRQ2E bit in IER is set to 1 in mode 1, 2, or 3 (EXPE = 1) with the WSCR set to 1, or in mode 2 and 3 (EXPE = 0), this pin is used as the $\overline{\text{IRQ2}}$. When TRGS1 and TRGS0 in ADCR of the A/D converter are both set to 1, this pin is used as the $\overline{\text{ADTRG}}$ input pin.

8.11 Port A

Port A is an 8-bit I/O port. Port A pins also function as keyboard buffer controller I/O pins, sense interrupt input pins, expansion A/D converter input pins, and address output pins. Functions change according to the operating mode. Port A input/output operates by Vcc independent from the Vcc power. Up to 5 V can be applied to port A pins if VccB power. Port A has the following registers. PADDR and PAPIN have the same address.

- Port A data direction register (PADDR)
- Port A output data register (PAODR)
- Port A input data register (PAPIN)

4	PA4DDR	0	W	<p>In mode 2 (EXPE = 1):</p> <p>The corresponding port A pins are addressed when PADDR bits are set to 1, and ignored when cleared to 0. The port A pins change the address I/O ports to output ports by setting the IOSE bit to 1.</p> <p>PA7 to PA2 pins are used as the keyboard controller I/O pins by setting the KBIO bits to 1, regardless of the operating mode, which is ignored according to PA7DDR to PA2DDR.</p> <p>PADDR has the same address as PAADDR. When port A status is returned.</p>
3	PA3DDR	0	W	
2	PA2DDR	0	W	
1	PA1DDR	0	W	
0	PA0DDR	0	W	

8.11.2 Port A Output Data Register (PAODR)

PAODR stores output data for port A.

Bit	Bit Name	Initial Value	R/W	Description
7	PA7ODR	0	R/W	PAODR can always be read or written regardless of the contents of PADDR.
6	PA6ODR	0	R/W	
5	PA5ODR	0	R/W	
4	PA4ODR	0	R/W	
3	PA3ODR	0	R/W	
2	PA2ODR	0	R/W	
1	PA1ODR	0	R/W	
0	PA0ODR	0	R/W	

4	PA4PIN	Undefined*	R
3	PA3PIN	Undefined*	R
2	PA2PIN	Undefined*	R
1	PA1PIN	Undefined*	R
0	PA0PIN	Undefined*	R

Note: * The initial value is determined according to the PA7 to PA0 pin states.

8.11.4 Pin Functions

- PA7/A23/ $\overline{\text{KIN15}}$ /CIN15/PS2CD

The pin function is switched as shown below according to the combination of operation modes, the KBIOE bit in KBCRH_2 of the keyboard buffer controller, the IOSE bit in SYSCON, and the PA7DDR bit.

Operating Mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE = 1)		
	KBIOE	0		1	0	
PA7DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA7 input pin	PA7 output pin	PS2CD output pin	PA7 input pin	A23 output pin	PA7 output pin
	$\overline{\text{KIN15}}$ input pin, CIN15 input pin, PS2CD input pin*					

Note: * When the KBIOE bit is set to 1 or the IICS bit in STCR is set to 1, this pin is an open-drain output, and has direct bus drive capability. This pin can always be used as the PS2CD, $\overline{\text{KIN15}}$, or CIN15 input pin.

PA6DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA6 input pin	PA6 output pin	PS2CC output pin	PA6 input pin	A22 output pin	PA6 output pin
	KIN14 input pin, CIN14 input pin, PS2CC input pin*					

Note: * When the KBIOE bit is set to 1 or the IICS bit in STCR is set to 1, this pin is open-drain output, and has direct bus drive capability. This pin can always be used as the PS2CC, KIN14, or CIN14 input pin.

- PA5/A21/ $\overline{\text{KIN13}}$ /CIN13/PS2BD

The pin function is switched as shown below according to the combination of operating mode, the KBIOE bit in KBCRH_1 of the keyboard buffer controller, the IOSE bit in SYSCR, and the PA5DDR bit.

Operating Mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE = 1)		
KBIOE	0		1	0		
PA5DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA5 input pin	PA5 output pin	PS2BD output pin	PA5 input pin	A21 output pin	PA5 output pin
	KIN13 input pin, CIN13 input pin, PS2BD input pin*					

Note: * When the KBIOE bit is set to 1 or the IICS bit in STCR is set to 1, this pin is open-drain output, and has direct bus drive capability. This pin can always be used as the PS2BD, $\overline{\text{KIN13}}$, or CIN13 input pin.

PA4DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA4 input pin	PA4 output pin	PS2BC output pin	PA4 input pin	A20 output pin	PA4 output pin
	KIN12 input pin, CIN12 input pin, PS2BC input pin*					

Note: * When the KBIOE bit is set to 1 or the IICS bit in STCR is set to 1, this pin is an open-drain output, and has direct bus drive capability. This pin can always be used as the PS2BC, KIN12, or CIN12 input pin.

- PA3/A19/ $\overline{\text{KIN11}}$ /CIN11/PS2AD

The pin function is switched as shown below according to the combination of operating mode, the KBIOE bit in KBCRH_0 of the keyboard buffer controller, the IOSE bit in SYSIOCR, and the PA3DDR bit.

Operating Mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE = 1)		
	KBIOE	0		1	0	
PA3DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA3 input pin	PA3 output pin	PS2AD output pin	PA3 input pin	A19 output pin	PA3 output pin
	KIN11 input pin, CIN11 input pin, PS2AD input pin*					

Note: * When the KBIOE bit is set to 1, this pin is an NMOS open-drain output, and has direct bus drive capability. This pin can always be used as the PS2AD, $\overline{\text{KIN11}}$, or CIN11 input pin.

PA2DDR	0	1	—	0	1	
IOSE	—	—	—	—	0	1
Pin Function	PA2 input pin	PA2 output pin	PS2AC output pin	PA2 input pin	A18 output pin	PA2 output pin
	KIN10 input pin, CIN10 input pin, PS2AC input pin*					

Note: * When the KBIOE bit is set to 1, this pin is an NMOS open-drain output, and does not have bus drive capability. This pin can always be used as the PS2AC, $\overline{\text{KIN10}}$, or CIN10 input pin.

- PA1/A17/ $\overline{\text{KIN9}}$ /CIN9

The pin function is switched as shown below according to the combination of operating mode and the IOSE bit in SYSCR and the PA1DDR bit.

Operating Mode	Modes 1, 2 (EXPE = 0), 3		Mode 2 (EXPE = 1)		
	PA1DDR	0	1	0	1
IOSE	—	—	—	0	1
Pin Function	PA1 input pin	PA1 output pin	PA1 input pin	A17 output pin	CIN9 input pin
	$\overline{\text{KIN9}}$ input pin, CIN9 input pin*				

Note: * This pin can always be used as the $\overline{\text{KIN9}}$ or CIN9 input pin.

Pin Function	PA0 input pin	PA0 output pin	PA0 input pin	A16 output pin	ou
	$\overline{\text{KIN8}}$ input pin, CIN8 input pin*				

Note: * This pin can always be used as the $\overline{\text{KIN8}}$ or CIN8 input pin.

8.11.5 Port A Input Pull-Up MOS

Port A has an on-chip input pull-up MOS function that can be controlled by software. The pull-up MOS function can be specified as on or off on a bit-by-bit basis.

The input pull-up MOS for pins PA7 to PA4 is always off when IICS is set to 1. When keyboard buffer control pin function is selected for pins PA7 to PA2, the input pull-up is always off.

Table 8.6 summarizes the input pull-up MOS states.

Table 8.6 Input Pull-Up MOS States (Port A)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Other Operation
1 to 3	Off	Off	On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, PADDR = 0, and PAODR = 1; otherwise off.

- Port B input data register (PBPIN)

8.12.1 Port B Data Direction Register (PBDDR)

PBDDR specifies input or output for the pins of port B on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	PB7DDR	0	W	PBDDR has the same address as P7I/O. When a read is performed on the PBDDR, the port 7 pin states will be returned.
6	PB6DDR	0	W	
5	PB5DDR	0	W	<ul style="list-style-type: none"> • Modes 1, 2, and 3 (EXPE = 1) When the ABW bit in WSCR is cleared, all port B pins automatically become input ports (D7 to D0), regardless of the direction indicated by PBDDR. When the ABW bit is 1, a port B pin becomes an output port if the corresponding PBDDR bit is set to 1, and an input port if the bit is cleared to 0. • Modes 2 and 3 (EXPE = 0) A port B pin becomes an output port if the corresponding PBDDR bit is set to 1, and an input port if the bit is cleared to 0.
4	PB4DDR	0	W	
3	PB3DDR	0	W	
2	PB2DDR	0	W	
1	PB1DDR	0	W	
0	PB0DDR	0	W	

4	PB4ODR	0	R/W
3	PB3ODR	0	R/W
2	PB2ODR	0	R/W
1	PB1ODR	0	R/W
0	PB0ODR	0	R/W

8.12.3 Port B Input Data Register (PBPIN)

PBPIN indicates the port B state.

Bit	Bit Name	Initial Value	R/W	Description
7	PB7PIN	Undefined*	R	Reading PBPIN always returns the pin state. If a write is performed, data will be written to the port B and the port 8 settings will change.
6	PB6PIN	Undefined*	R	
5	PB5PIN	Undefined*	R	
4	PB4PIN	Undefined*	R	
3	PB3PIN	Undefined*	R	
2	PB2PIN	Undefined*	R	
1	PB1PIN	Undefined*	R	
0	PB0PIN	Undefined*	R	

Note: * The initial value is determined according to the PB7 to PB0 pin states.

ADW	0	1	1	0	
PBnDDR	—	0	1	0	
Pin Function	Dn I/O pin	PBn input pin	PBn output pin	PBn input pin	
		$\overline{WUE_n}$ input pin* ¹			

Notes: 1. Except when used as a data bus pin, this pin can always be used as the $\overline{WUE_n}$ pin. (n = 7 to 4)

2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

- PB3/D3/ $\overline{WUE3}$ *²/ $\overline{CS4}$

The pin function is switched as shown below according to the combination of the mode, the HI12E and CS4E bits in SYSCR2, the ABW bit in WSCR, and the PB3

Operating Mode	Mode 1 and Modes 2, 3 (EXPE = 1)			Modes 2, 3 (EXPE = 0)	
HI12E	—			Either cleared to 0	
CS4E	—				
ABW	0	1		—	
PB3DDR	—	0	1	0	1
Pin Function	D3 I/O pin	PB3 input pin	PB3 output pin	PB3 input pin	PB3 output pin
		$\overline{WUE3}$ input pin* ¹			

Notes: 1. Except when used as a data bus pin, this pin can always be used as the $\overline{WUE3}$ pin. The $\overline{CS4}$ input pin can only be used in mode 2 or 3 (EXPE = 0).

2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

- PB2/D2/ $\overline{WUE2}$ *²/ $\overline{CS3}$

The pin function is switched as shown below according to the combination of the mode, the HI12E and CS3E bits in SYSCR2, the ABW bit in WSCR, and the PB2

- Notes: 1. Except when used as a data bus pin, this pin can always be used as the \overline{WU} pin. The $\overline{CS3}$ input pin can only be used in mode 2 or 3 (EXPE = 0).
2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

- PB1/D1/ $\overline{WUE1}$ /HIRQ4/LSCI*4

The pin function is switched as shown below according to the combination of the operating mode, the HI12E and CS4E bits in SYSCR2, the LSCIE bits in HICR0 of host interface, the ABW bit in WSCR, and the PB1DDR bit.

Operating Mode	Modes 1, 2, 3 (EXPE = 1)			Mode 2, 3 (EXPE = 0)		
LSCIE	0*3			0		
HI12E	—			Either cleared to 0		1
CS4E	—					1
ABW	0	1		—		—
PB1DDR	—	0	1	0	1	—
Pin Function	D1 I/O pin	PB1 input pin	PB1 output pin	PB1 input pin	PB1 output pin	HIRQ4 output pin
		LSCI input pin*2				
WUE1 input pin*2*4						

- Notes: 1. When bit LSCIE is set to 1 in HICR0, bits HI12E and PB1DDR should be cleared to 0.
2. Except when used as a data bus pin, this pin can always be used as the \overline{WU} pin. The HIRQ4 output pin and LSCI I/O pin can only be used in mode 2 or 3 (EXPE = 0).
3. In mode 1, 2, 3 (EXPE = 1), clear the LSCIE bit to 0.
4. Not supported by the H8S/2148B and H8S/2145B (5-V version).

HI12E	—			Either cleared to 0		1
CS3E	—					1
ABW	0	1		—		—
PB0DDR	—	0	1	0	1	—
Pin Function	D0 I/O pin	PB0 input pin	PB0 output pin	PB0 input pin	PB0 output pin	HIRQ3 output pin
				LSM \bar{I} input pin ^{*2}		
		WUE0 input pin ^{*2}				

- Notes:
1. When bit LSMIE is set to 1 in HICR0, bits HI12E and PB0DDR should be cleared to 0.
 2. Except when used as a data bus pin, this pin can always be used as the \bar{V}_{DD} pin. The HIRQ3 output pin and $\bar{L}SM\bar{I}$ I/O pin can only be used in mode 2 (output mode 0).
 3. In mode 1, 2, 3 (EXPE = 1), clear the LSMIE bit to 0.
 4. Not supported by the H8S/2148B and H8S/2145B (5-V version).

Table 8.7 Input Pull-Up MOS States (Port B)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Ope
1, 2, 3 (EXPE = 1) with ABW in WSCR = 0	Off	Off	Off	Off
1, 2, 3 (EXPE = 1) with ABW in WSCR = 1, or 2, 3 (EXPE = 0)			On/Off	On/O

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when the pin is in the input state, PBDDR = 0, and PBODR = 1; otherwise off.

8.13 Additional Overview for H8S/2160B and H8S/2161B

The H8S/2160B and H8S/2161B has fifteen I/O ports (ports 1 to 6, 8, 9, A to G), and only port (port 7).

Table 8.8 is a summary of the additional port functions. As the functions of ports 1 to 9 are the same on the H8S/2140B, H8S/2141B, H8S/2148B, and H8S/2145B, table 8.1 provides a summary.

Each extra port includes a data direction register (DDR) that controls input/output, and output registers (ODR) for storing output data.

Ports C to E, and F have an on-chip input pull-up MOS function. On ports C to F, whether input pull-up MOS is on or off is controlled by the corresponding DDR and ODR.

Ports C to F, and G can drive a single-TTL load and 30-pF-capacitive load. All I/O ports are capable of driving a Darlington transistor when they are in output.

Port C	8-bit I/O port	PC7 to PC0	On- pull-
Port D	8-bit I/O port	PD7 to PD0	On- pull-
Port E	8-bit I/O port	PE7 to PE0	On- pull-
Port F	8-bit I/O port	PF7 to PF0	On- pull-
Port G	8-bit I/O port	PG7 to PG0	

8.14 Ports C, D

Port C and port D are two sets of 8-bit I/O ports. The pin functions are the same in all modes.

- Port C data direction register (PCDDR)
- Port C output data register (PCODR)
- Port C input data register (PCPIN)
- Port C Nch-OD control register (PCNOCR)
- Port D data direction register (PDDDR)
- Port D output data register (PDODR)
- Port D input data register (PDPIN)
- Port D Nch-OD control register (PDNOCR)

4	PC4DDR	0	W
3	PC3DDR	0	W
2	PC2DDR	0	W
1	PC1DDR	0	W
0	PC0DDR	0	W

read, the port C pin states will be returned.

Bit	Bit Name	Initial Value	R/W	Description
7	PD7DDR	0	W	0: Port D pin is an input pin
6	PD6DDR	0	W	1: Port D pin is an output pin
5	PD5DDR	0	W	PDDDR has the same address as PDDR. When read, the port D pin states will be returned.
4	PD4DDR	0	W	
3	PD3DDR	0	W	
2	PD2DDR	0	W	
1	PD1DDR	0	W	
0	PD0DDR	0	W	

4	PC4ODR	0	R/W
3	PC3ODR	0	R/W
2	PC2ODR	0	R/W
1	PC1ODR	0	R/W
0	PC0ODR	0	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PD7ODR	0	R/W	PDODR can always be read or written regardless of the contents of PDDDR.
6	PD6ODR	0	R/W	
5	PD5ODR	0	R/W	
4	PD4ODR	0	R/W	
3	PD3ODR	0	R/W	
2	PD2ODR	0	R/W	
1	PD1ODR	0	R/W	
0	PD0ODR	0	R/W	

4	PC4PIN	Undefined*	R
3	PC3PIN	Undefined*	R
2	PC2PIN	Undefined*	R
1	PC1PIN	Undefined*	R
0	PC0PIN	Undefined*	R

Note: * The initial value is determined according to the PC7 to PC0 pin states.

Bit	Bit Name	Initial Value	R/W	Description
7	PD7PIN	Undefined*	R	PDPIN indicates the port D state. PDPIN is located at the same address as PDDDR. If a write is performed to PDPIN, the port D settings will change.
6	PD6PIN	Undefined*	R	
5	PD5PIN	Undefined*	R	
4	PD4PIN	Undefined*	R	
3	PD3PIN	Undefined*	R	
2	PD2PIN	Undefined*	R	
1	PD1PIN	Undefined*	R	
0	PD0PIN	Undefined*	R	

Note: * The initial value is determined according to the PD7 to PD0 pin states.

5	PC5NOCR	0	R/W
4	PC4NOCR	0	R/W
3	PC3NOCR	0	R/W
2	PC2NOCR	0	R/W
1	PC1NOCR	0	R/W
0	PC0NOCR	0	R/W

disabled)

Bit	Bit Name	Initial Value	R/W	Description
7	PD7NOCR	0	R/W	0: CMOS (p-channel driver enabled)
6	PD6NOCR	0	R/W	1: N-channel open drain (p-channel driver disabled)
5	PD5NOCR	0	R/W	
4	PD4NOCR	0	R/W	
3	PD3NOCR	0	R/W	
2	PD2NOCR	0	R/W	
1	PD1NOCR	0	R/W	
0	PD0NOCR	0	R/W	

8.14.5 Pin Functions

DDR	0		1		
	—		0		0
NOCR	—		0		0
ODR	0	1	0	1	0
N-ch. driver	OFF		ON	OFF	ON
P-ch. driver	OFF		OFF	ON	ON
Input pull-up MOS	OFF	ON	OFF		
Pin function	Input pin		Output pin		

Rev. 3.00 Mar 21, 2006 pag

REJ03

RENESAS

Mode	Reset	Hardware Standby Mode	Software Standby Mode	Other Operati
1 to 3	Off	Off	On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when PCDDR = 0 and PCODR = 1 (PDDDR = 0 and PDODR = 1); otherwise

8.15 Ports E, F

Port E and port F are two sets of 8-bit I/O ports. The pins of ports E and F have the same function in all operating modes.

- Port E data direction register (PEDDR)
- Port E output data register (PEODR)
- Port E input data register (PEPIN)
- Port E Nch-OD control register (PENOCR)
- Port F data direction register (PFDDR)
- Port F output data register (PFODR)
- Port F input data register (PFPIN)
- Port F Nch-OD control register (PFNOCR)

4	PE4DDR	0	W
3	PE3DDR	0	W
2	PE2DDR	0	W
1	PE1DDR	0	W
0	PE0DDR	0	W

Bit	Bit Name	Initial Value	R/W	Description
7	PF7DDR	0	W	0: Port F pin is an input pin
6	PF6DDR	0	W	1: Port F pin is an output pin
5	PF5DDR	0	W	PFDDR has the same address as PFD read, the port F pin states will be returned
4	PF4DDR	0	W	
3	PF3DDR	0	W	
2	PF2DDR	0	W	
1	PF1DDR	0	W	
0	PF0DDR	0	W	

4	PE4ODR	0	R/W
3	PE3ODR	0	R/W
2	PE2ODR	0	R/W
1	PE1ODR	0	R/W
0	PE0ODR	0	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PF7ODR	0	R/W	PFODR can always be read or written regardless of the contents of PFDDR.
6	PF6ODR	0	R/W	
5	PF5ODR	0	R/W	
4	PF4ODR	0	R/W	
3	PF3ODR	0	R/W	
2	PF2ODR	0	R/W	
1	PF1ODR	0	R/W	
0	PF0ODR	0	R/W	

4	PE4PIN	Undefined*	R
3	PE3PIN	Undefined*	R
2	PE2PIN	Undefined*	R
1	PE1PIN	Undefined*	R
0	PE0PIN	Undefined*	R

Note: * The initial value is determined according to the PE7 to PE0 pin states.

Bit	Bit Name	Initial Value	R/W	Description
7	PF7PIN	Undefined*	R	PFPIN indicates the port F state. PFPIN has the same address as PFDDR. If a write is made to the port F settings will change.
6	PF6PIN	Undefined*	R	
5	PF5PIN	Undefined*	R	
4	PF4PIN	Undefined*	R	
3	PF3PIN	Undefined*	R	
2	PF2PIN	Undefined*	R	
1	PF1PIN	Undefined*	R	
0	PF0PIN	Undefined*	R	

Note: * The initial value is determined according to the PF7 to PF0 pin states.

5	PE5NOCR	0	R/W
4	PE4NOCR	0	R/W
3	PE3NOCR	0	R/W
2	PE2NOCR	0	R/W
1	PE1NOCR	0	R/W
0	PE0NOCR	0	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PF7NOCR	0	R/W	0: CMOS (p-channel driver enabled)
6	PF6NOCR	0	R/W	1: N-channel open drain (p-channel driver disabled)
5	PF5NOCR	0	R/W	
4	PF4NOCR	0	R/W	
3	PF3NOCR	0	R/W	
2	PF2NOCR	0	R/W	
1	PF1NOCR	0	R/W	
0	PF0NOCR	0	R/W	

8.15.5 Pin Functions

DDR	0		1		
	—		0		1
NOCR	—		0		1
ODR	0	1	0	1	0
N-ch. driver	OFF		ON	OFF	ON
P-ch. driver	OFF		OFF	ON	OFF
Input pull-up MOS	OFF	ON	OFF		
Pin function	Input pin		Output pin		

Mode	Reset	Hardware Standby Mode	Software Standby Mode	Other Operate
1 to 3	Off	Off	On/Off	On/Off

Legend:

Off: Input pull-up MOS is always off.

On/Off: On when PEDDR = 0 and PEODR = 1 (PFDDR = 0 and PFODR = 1); otherwise

8.16 Port G

Port G is an 8-bit I/O port. Port G pin functions are the same in all operating modes. The type of port G is NMOS open-drain.

- Port G data direction register (PGDDR)
- Port G output data register (PGODR)
- Port G input data register (PGPIN)
- Port G Nch-OD control register (PGNOCR)

4	PG4DDR	0	W
3	PG3DDR	0	W
2	PG2DDR	0	W
1	PG1DDR	0	W
0	PG0DDR	0	W

read, the port G pin states will be returned.

8.16.2 Port G Output Data Register (PGODR)

PGODR stores output data for the pins on port G.

Bit	Bit Name	Initial Value	R/W	Description
7	PG7ODR	0	R/W	PGODR can always be read or written regardless of the contents of PGDDR.
6	PG6ODR	0	R/W	
5	PG5ODR	0	R/W	
4	PG4ODR	0	R/W	
3	PG3ODR	0	R/W	
2	PG2ODR	0	R/W	
1	PG1ODR	0	R/W	
0	PG0ODR	0	R/W	

4	PG4PIN	Undefined*	R
3	PG3PIN	Undefined*	R
2	PG2PIN	Undefined*	R
1	PG1PIN	Undefined*	R
0	PG0PIN	Undefined*	R

Note: * The initial value is determined according to the PG7 to PG0 pin states.

8.16.4 Port G Nch-OD Control Register (PGNOCR)

PGNOCR specifies the output driver type for pins on port G which are configured as output on a bit-by-bit basis.

Bit	Bit Name	Initial Value	R/W	Description
7	PG7NOCR	0	R/W	0: NMOS push-pull (Vcc-side n-channel driver enabled)
6	PG6NOCR	0	R/W	
5	PG5NOCR	0	R/W	1: Vss-side N-channel open drain (Vcc-side n-channel driver disabled)
4	PG4NOCR	0	R/W	
3	PG3NOCR	0	R/W	
2	PG2NOCR	0	R/W	
1	PG1NOCR	0	R/W	
0	PG0NOCR	0	R/W	

- Operable at a maximum carrier frequency of 625 kHz using pulse division (at 10 M operation)
- Duty cycles from 0 to 100% with 1/256 resolution (100% duty realized by port out)
- Direct or inverted PWM output, and PWM output enable/disable control

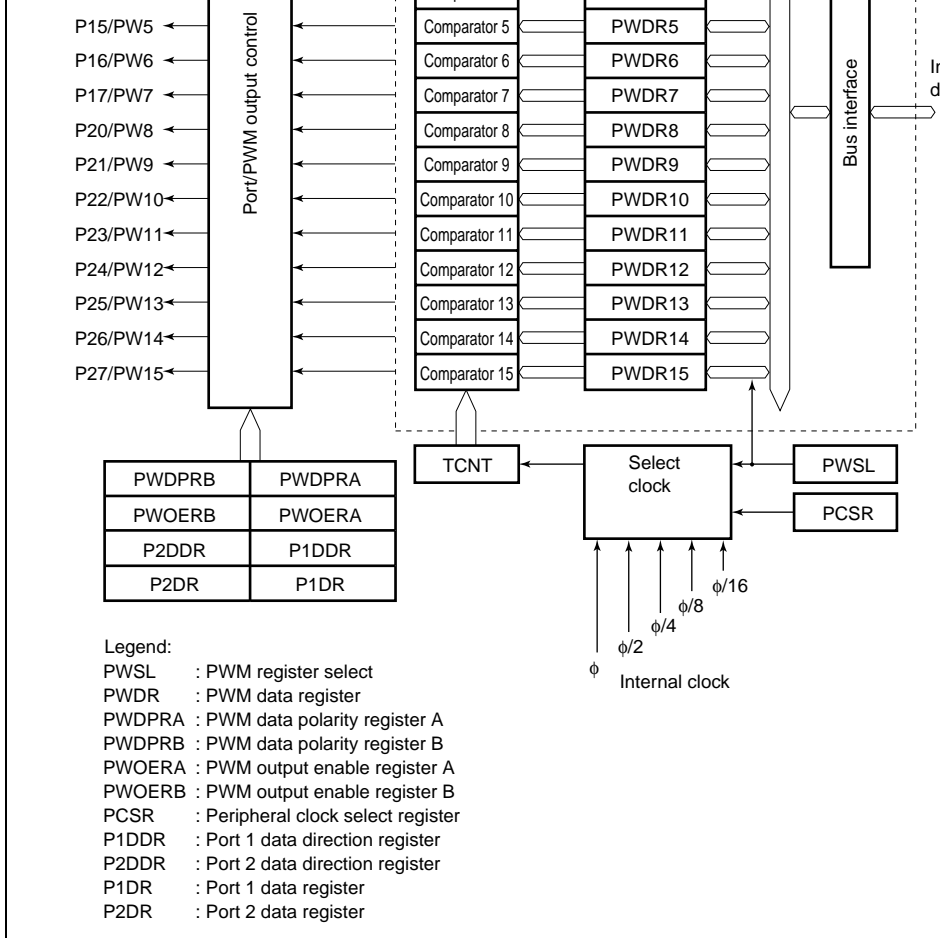


Figure 9.1 Block Diagram of PWM Timer

9.3 Register Descriptions

The PWM has the following registers. To access PCSR, the FLSHE bit in the serial timer control register (STCR) must be cleared to 0. For details on the serial timer control register (STCR), see section 3.2.3, Serial Timer Control Register (STCR).

- PWM register select (PWSL)
- PWM data registers 0 to 15 (PWDR0 to PWDR15)
- PWM data polarity register A (PWDPRB)
- PWM data polarity register B (PWDPRB)
- PWM output enable register A (PWOERA)
- PWM output enable register B (PWOERB)
- Peripheral clock select register (PCSR)

PCSR, select the internal clock input to TCN
 PWM. For details, see table 9.2.

The resolution, PWM conversion period, and
 frequency depend on the selected internal cl
 can be obtained from the following equations

Resolution (minimum pulse width) = 1/intern
 frequency

PWM conversion period = resolution × 256

Carrier frequency = 16/PWM conversion per

With a 10 MHz system clock (ϕ), the resoluti
 conversion period, and carrier frequency are
 in table 9.3.

5	—	1	R	Reserved This bit is always read as 1 and cannot be m
4	—	0	R	Reserved This bit is always read as 0 and cannot be m

0100: PWDR4 selected
 0101: PWDR5 selected
 0110: PWDR6 selected
 0111: PWDR7 selected
 1000: PWDR8 selected
 1001: PWDR9 selected
 1010: PWDR10 selected
 1011: PWDR11 selected
 1100: PWDR12 selected
 1101: PWDR13 selected
 1110: PWDR14 selected
 1111: PWDR15 selected

Table 9.2 Internal Clock Selection

PWCKE	PWSL		PCSR		Description
	PWCKS	PWCKB	PWCKA		
0	—	—	—		Clock input is disabled
1	0	—	—		ϕ (system clock) is selected
					$\phi/2$ is selected
	1	0	0		$\phi/4$ is selected
					$\phi/8$ is selected
					$\phi/16$ is selected

9.3.2 PWM Data Registers (PWDR0 to PWDR15)

PWDR are 8-bit readable/writable registers. The PWM has sixteen PWM data registers. PWDR specifies the duty cycle of the basic pulse to be output, and the number of additional pulses. The value set in PWDR corresponds to a 0 or 1 ratio in the conversion period. The four bits specify the duty cycle of the basic pulse as 0/16 to 15/16 with a resolution of 1/16. The lower four bits specify how many extra pulses are to be added within the conversion period comprising 16 basic pulses. Thus, a specification of 0/256 to 255/256 is possible for 0% to 100% within the conversion period. For 256/256 (100%) output, port output should be used. PWDR15 are initialized to H'00.

9.3.3 PWM Data Polarity Registers A and B (PWPRA, PWPBR)

Each PWPDR selects the PWM output phase.

- PWPRA

Bit	Bit Name	Initial Value	R/W	Description
7	OS7	0	R/W	Output Select 7 to 0
6	OS6	0	R/W	These bits select the PWM output phase. Bit 6 and bit 5 OS0 correspond to outputs PW7 to PW0.
5	OS5	0	R/W	
4	OS4	0	R/W	0: PWM direct output (PWDR value corresponds to the width of output)
3	OS3	0	R/W	1: PWM inverted output (PWDR value corresponds to the low width of output)
2	OS2	0	R/W	
1	OS1	0	R/W	
0	OS0	0	R/W	

2	OS10	0	R/W	1: PWM inverted output (PWDR value correct) low width of output)
1	OS9	0	R/W	
0	OS8	0	R/W	

9.3.4 PWM Output Enable Registers A and B (PWOERA, PWOERB)

Each PWOER switches between PWM output and port output.

- PWOERA

Bit	Bit Name	Initial Value	R/W	Description
7	OE7	0	R/W	Output Enable 7 to 0
6	OE6	0	R/W	These bits, together with P1DDR, specify the pin state. Bits OE7 to OE0 correspond to OE7 to OE0.
5	OE5	0	R/W	
4	OE4	0	R/W	P1nDDR OEn: Pin state
3	OE3	0	R/W	0X: Port input
2	OE2	0	R/W	10: Port output or PWM 256/256 output
1	OE1	0	R/W	11: PWM output (0 to 255/256 output)
0	OE0	0	R/W	

Legend:

X: Don't care

2	OE10	0	R/W	0X: Port input
1	OE9	0	R/W	10: Port output or PWM 256/256 output
0	OE8	0	R/W	11: PWM output (0 to 255/256 output)

Legend:

X: Don't care

To perform PWM 256/256 output when DDR = 1 and OE = 0, the corresponding pin should be set as port output. The corresponding pin can be set as port output in single-chip mode or v = 1 and CS256E = 0 in SYSCR in extended mode with on-chip ROM. Otherwise, it should be noted that an address bus is output to the corresponding pin.

DR data is output when the corresponding pin is used as port output. A value corresponding to PWM 256/256 output is determined by the OS bit, so the value should have been set to OS beforehand.

9.3.5 Peripheral Clock Select Register (PCSR)

PCSR selects the PWM input clock.

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The initial value cannot be changed.
2	PWCKB	0	R/W	PWM Clock Select B, A
1	PWCKA	0	R/W	Together with bits PWCKE and PWCKS in PCSR, these bits select the internal clock input to the PWM. For details, see table 9.2.
0	—	0	R	Reserved This bit is always read as 0. The initial value cannot be changed.

0001	
0010	
0011	
0100	
0101	
0110	
0111	
1000	
1001	
1010	
1011	
1100	
1101	
1110	
1111	

The lower four bits of PWDR specify the position of pulses added to the 16 basic pulses. Each additional pulse adds a high period (when OS = 0) with a width equal to the resolution of the rising edge of a basic pulse. When the upper four bits of PWDR are 0000, there is no pulse added to the basic pulse, but the timing for adding pulses is the same. Table 9.5 shows the position of the basic pulse, but the timing for adding pulses is the same. Table 9.5 shows the position of the basic pulse, but the timing for adding pulses is the same.

0001								Yes						
0011								Yes				Yes		
0100				Yes				Yes				Yes		
0101				Yes				Yes				Yes		Yes
0110				Yes		Yes		Yes				Yes		Yes
0111				Yes		Yes		Yes		Yes		Yes		Yes
1000		Yes		Yes		Yes		Yes		Yes		Yes		Yes
1001		Yes		Yes		Yes		Yes		Yes		Yes		Yes
1010		Yes		Yes		Yes	Yes	Yes		Yes		Yes		Yes
1011		Yes		Yes		Yes	Yes	Yes		Yes	Yes	Yes		Yes
1100		Yes	Yes	Yes		Yes	Yes	Yes		Yes	Yes	Yes		Yes
1101		Yes	Yes	Yes		Yes	Yes	Yes		Yes	Yes	Yes	Yes	Yes
1110		Yes	Yes	Yes	Yes	Yes	Yes	Yes		Yes	Yes	Yes	Yes	Yes
1111		Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

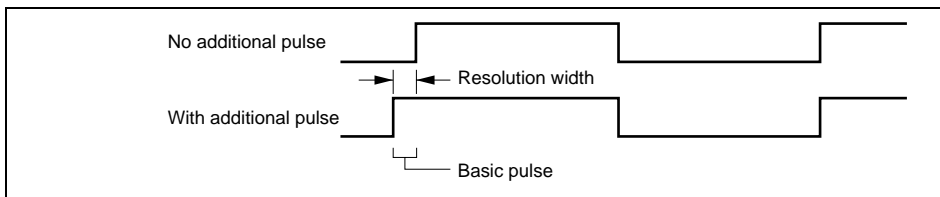


Figure 9.2 Example of Additional Pulse Timing (when Upper 4 Bits of PWDR)

- Two resolution settings
The resolution can be set equal to one or two system clock cycles.
- Two base cycle settings
The base cycle can be set equal to $T \times 64$ or $T \times 256$, where T is the resolution.
- Four operating speeds
- Four operation clocks (by combination of two resolution settings and two base cycle

Figure 10.1 shows a block diagram of the PWM (D/A) module.

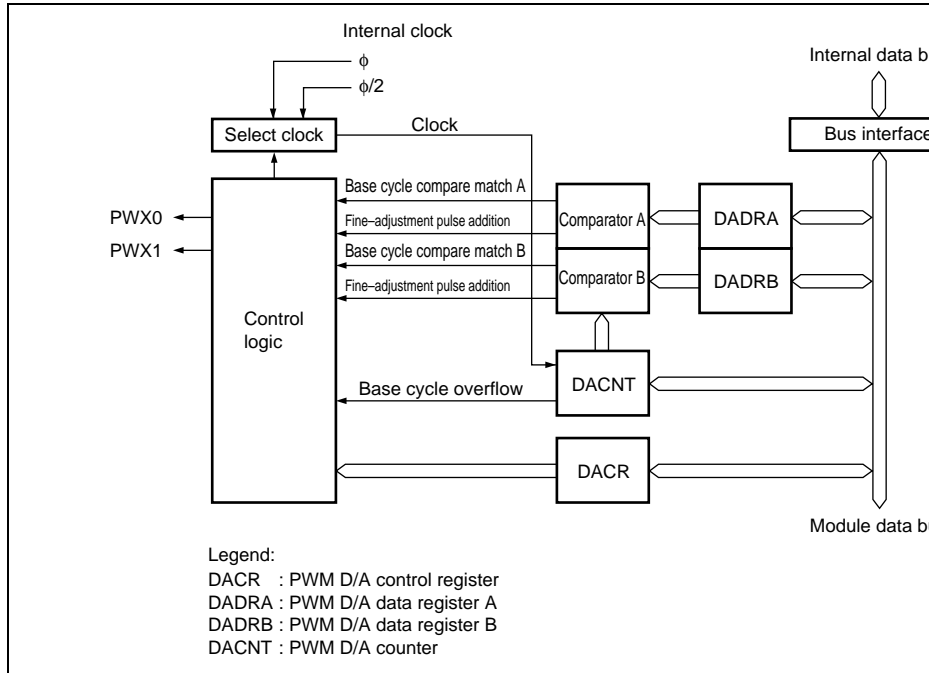


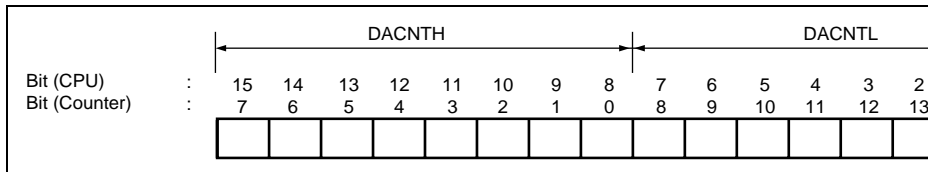
Figure 10.1 PWM (D/A) Block Diagram

10.3 Register Descriptions

The PWM (D/A) module has the following registers. The PWM (D/A) registers are assigned the same addresses with other registers. The registers are selected by the IICE bit in the serial timer control register (STCR). For details on STCR, see section 3.2.3, Serial Timer Control Register (STCR).

- PWM (D/A) counter H (DACNTH)
- PWM (D/A) counter L (DACNTL)
- PWM (D/A) data register AH (DADRAH)
- PWM (D/A) data register AL (DADRAL)
- PWM (D/A) data register BH (DADRBH)
- PWM (D/A) data register BL (DADRBL)
- PWM (D/A) control register (DACR)

Note: The same addresses are shared by DADRA and DACR, and by DADRB and DADRBL. Switching is performed by the REGS bit in DACNT or DADRB.



- DACNTH

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	UC7 UC0	All 0	R/W	Upper Up-Counter

- DACNTL

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	UC8 UC13	All 0	R/W	Lower Up-Counter
1	—	1	R	Reserved This bit is always read as 1 and cannot be
0	REGS	1	R/W	Register Select DADRA and DACR, and DADRB and DACR located at the same addresses. The REGS which registers can be accessed. 0: DADRA and DADRB can be accessed 1: DACR and DACNT can be accessed

15	DA13	1	R/W	D/A Data 13 to 0
14	DA12	1	R/W	These bits set a digital value to be converted to analog value.
13	DA11	1	R/W	
12	DA10	1	R/W	In each base cycle, the DACNT value is compared with the DADR value to determine the period of the output waveform, and to decide the period of the output waveform.
11	DA9	1	R/W	
10	DA8	1	R/W	output a fine-adjustment pulse equal in width to the period of the output waveform.
9	DA7	1	R/W	resolution. To enable this operation, this register must be set within a range that depends on the CF register value.
8	DA6	1	R/W	DADR value is outside this range, the PWM period is held constant.
7	DA5	1	R/W	
6	DA4	1	R/W	A channel can be operated with 12-bit precision by setting the two lowest data bits (DA1 and DA0) to 1.
5	DA3	1	R/W	keeping the two lowest data bits (DA1 and DA0) to 0. The two lowest data bits correspond to the two lowest bits in DACNT.
4	DA2	1	R/W	
3	DA1	1	R/W	
2	DA0	1	R/W	
1	CFS	1	R/W	Carrier Frequency Select 0: Base cycle = resolution (T) × 64 DADR range = H'0401 to H'FFFD 1: Base cycle = resolution (T) × 256 DADR range = H'0103 to H'FFFF
0	—	1	R	Reserved This bit is always read as 1 and cannot be modified.

10	DA8	1	R/W	output a fine-adjustment pulse equal in width to the resolution. To enable this operation, this register must be set within a range that depends on the CFS bit. If the DADR value is outside this range, the PWM output is held constant.
9	DA7	1	R/W	
8	DA6	1	R/W	
7	DA5	1	R/W	
6	DA4	1	R/W	A channel can be operated with 12-bit precision by setting the two lowest data bits (DA1 and DA0) to 0. The two lowest data bits correspond to the two highest bits in DACNT.
5	DA3	1	R/W	
4	DA2	1	R/W	
3	DA1	1	R/W	
2	DA0	1	R/W	
1	CFS	1	R/W	Carrier Frequency Select 0: Base cycle = resolution (T) × 64 DADR range = H'0401 to H'FFFD 1: Base cycle = resolution (T) × 256 DADR range = H'0103 to H'FFFF
0	REGS	1	R/W	Register Select DADRA and DACR, and DADRB and DACNT are located at the same addresses. The REGS bit selects which registers can be accessed. 0: DADRA and DADRB can be accessed 1: DACR and DACNT can be accessed

Normally this bit should be cleared to 0.
 0: PWM (D/A) in user state: Normal operation
 1: PWM (D/A) in test state: Correct conversion unobtainable

6	PWME	0	R/W	PWM Enable Starts or stops the PWM D/A counter (DACNT) 0: DACNT operates as a 14-bit up-counter 1: DACNT halts at H'0003
5, 4	—	All 1	R	Reserved These bits are always read as 1 and cannot be modified.
3	OEB	0	R/W	Output Enable B Enables or disables output on PWM (D/A) channel B 0: PWM (D/A) channel B output (at the PWBX) disabled 1: PWM (D/A) channel B output (at the PWBX) enabled

Selects the phase of the PWM (D/A) output

0: Direct PWM (D/A) output

1: Inverted PWM (D/A) output

0	CKS	0	R/W	Clock Select
---	-----	---	-----	--------------

Selects the PWM (D/A) resolution. If the system clock (ϕ) frequency is 10 MHz, resolutions of 100 ns, can be selected.

0: Operates at resolution (T) = system clock (t_{cyc})

1: Operates at resolution (T) = system clock (t_{cyc}) \times 2

when the lower byte is written to, the lower-byte write data and TEMP value are combined. When the upper byte is written to, the upper-byte write data and TEMP value are combined. The combined 16-bit value is written in the register.

Read: When the upper byte is read from, the upper-byte value is transferred to the CPU. When the lower byte is read from, the lower-byte value is transferred to TEMP. Next, when the lower byte is read from, the lower-byte value in TEMP is transferred to the CPU.

These registers should always be accessed 16 bits at a time with a MOV instruction, and the upper byte should always be accessed before the lower byte. Correct data will not be transferred if only the upper byte or only the lower byte is accessed. Also note that a bit manipulation instruction cannot be used to access these registers.

Example 1: Write to DACNT

```
MOV.W R0, @DACNT ; Write R0 contents to DACNT
```

Example 2: Read DADRA

```
MOV.W @DADRA, R0 ; Copy contents of DADRA to R0
```

Table 10.2 Read and Write Access Methods for 16-Bit Registers

Register Name	Read		Write	
	Word	Byte	Word	Byte
DADRA and DADRB	Yes	Yes	Yes	×
DACNT	Yes	×	Yes	×

Legend:

Yes: Permitted type of access. Word access includes successive byte accesses to the upper (first) and lower byte (second).

×: This type of access may give incorrect results.

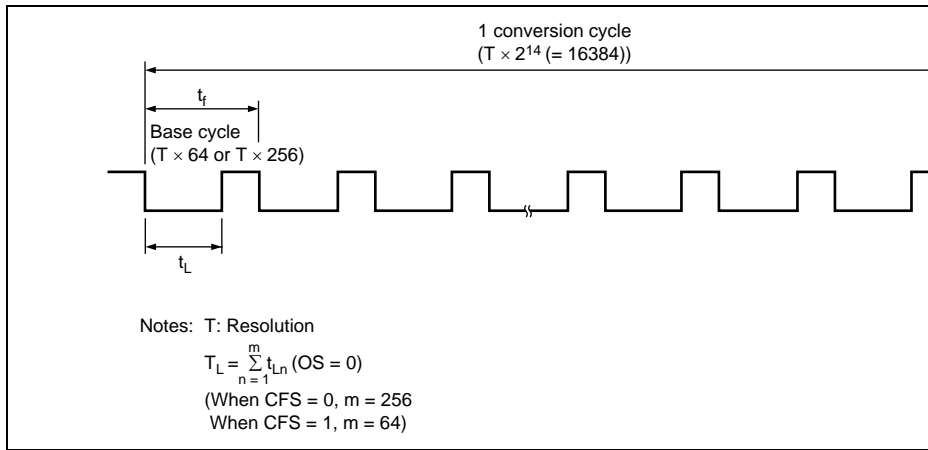


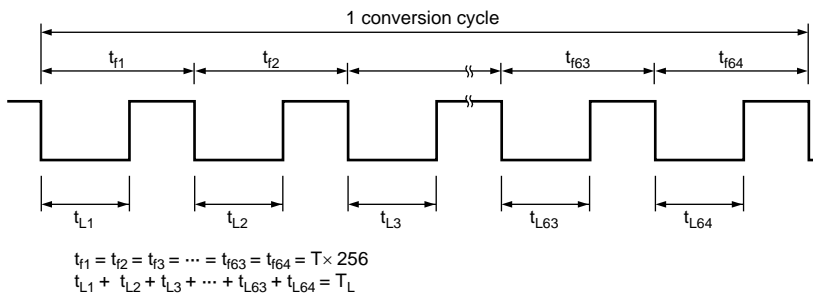
Figure 10.2 PWM D/A Operation

0	0.1	0	6.4	1638.4	1. Always low (or high) (DADR = H'0001 to H'03FD) 2. (Data value) × T (DADR = H'0401 to H'FFFD)	14				
						12			0	0
						10	0	0	0	0
	1	25.6	1. Always low (or high) (DADR = H'0003 to H'00FF) 2. (Data value) × T (DADR = H'0103 to H'FFFF)		14					
					12			0	0	
					10	0	0	0	0	
1	0.2	0	12.8	3276.8	1. Always low (or high) (DADR = H'0001 to H'03FD) 2. (Data value) × T (DADR = H'0401 to H'FFFD)	14				
						12			0	0
						10	0	0	0	0
	1	51.2	1. Always low (or high) (DADR = H'0003 to H'00FF) 2. (Data value) × T (DADR = H'0103 to H'FFFF)		14					
					12			0	0	
					10	0	0	0	0	

Note: * This column indicates the conversion cycle when specific DADR bits are fixe

$$t_{L1} + t_{L2} + t_{L3} + \dots + t_{L255} + t_{L256} = T_L$$

a. CFS = 0 [base cycle = resolution (T) × 64]

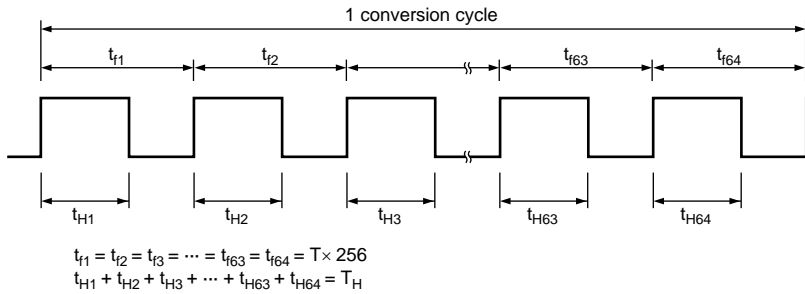


b. CFS = 1 [base cycle = resolution (T) × 256]

Figure 10.3 Output Waveform (OS = 0, DADR Corresponds to T_L)

$$t_{H1} + t_{H2} + t_{H3} + \dots + t_{H255} + t_{H256} = T_H$$

a. CFS = 0 [base cycle = resolution (T) × 64]



b. CFS = 1 [base cycle = resolution (T) × 256]

Figure 10.4 Output Waveform (OS = 1, DADR Corresponds to T_H)

An example of setting CFS to 1 (basic cycle = resolution (T) × 256) and OS to 1 (PWM output) is shown as an additional pulse. When CFS is set to 1, the duty ratio of the basic pulse is determined by the upper eight bits (DA13 to DA6) in DADR, and the position of the additional pulse is determined by the following six bits (DA5 to DA0) as shown in figure 10.5.

Table 10.4 shows the position of the additional pulse.

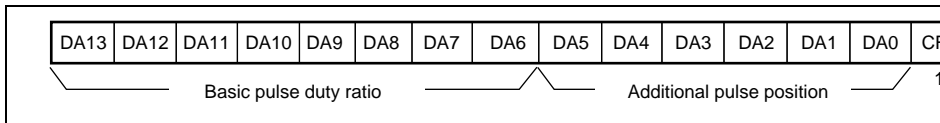


Figure 10.5 D/A Data Register Configuration when CFS = 1

Here, the case of DADR = H'0207 (B'0000 0010 0000 0111) is considered. Figure 10.6 shows the output waveform. Because CFS = 1 and the value of upper eight bits is B'0000 0010, the duty ratio of the basic pulse is $2/256 \times (T)$ of high width.

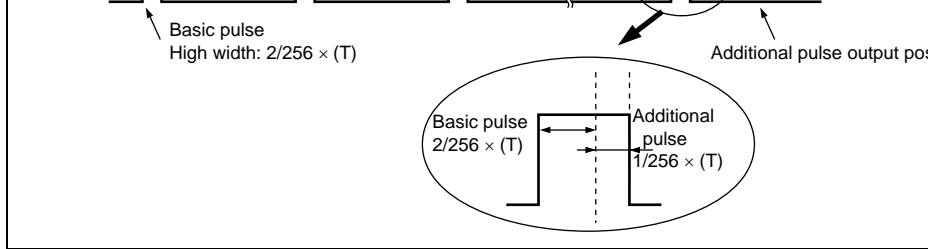


Figure 10.6 Output Waveform when DADR = H'0207 (OS = 1)

Note that the case of CFS = 0 (basic cycle = resolution (T) × 64) is similar other than the position of the basic pulse is determined by the upper six bits, and the position of the additional pulse is determined by the following eight bits.

- Selection of four clock sources
One of the three internal clocks ($\phi/2$, $\phi/8$, or $\phi/32$), or an external clock input can be selected (enabling use as an external event counter).
- Two independent comparators
Two independent waveforms can be output.
- Four independent input capture channels
The rising or falling edge can be selected.
Buffer modes can be specified.
- Counter clearing
The free-running counters can be cleared on compare-match A.
- Seven independent interrupts
Two compare-match interrupts, four input capture interrupts, and one overflow interrupt can be requested independently.
- Special functions provided by automatic addition function
The contents of OCRAR and OCRAF can be added to the contents of OCRA automatically, enabling a periodic waveform to be generated without software intervention. The contents of ICRD can be added automatically to the contents of OCRDM $\times 2$, enabling input capture operations in this interval to be restricted.

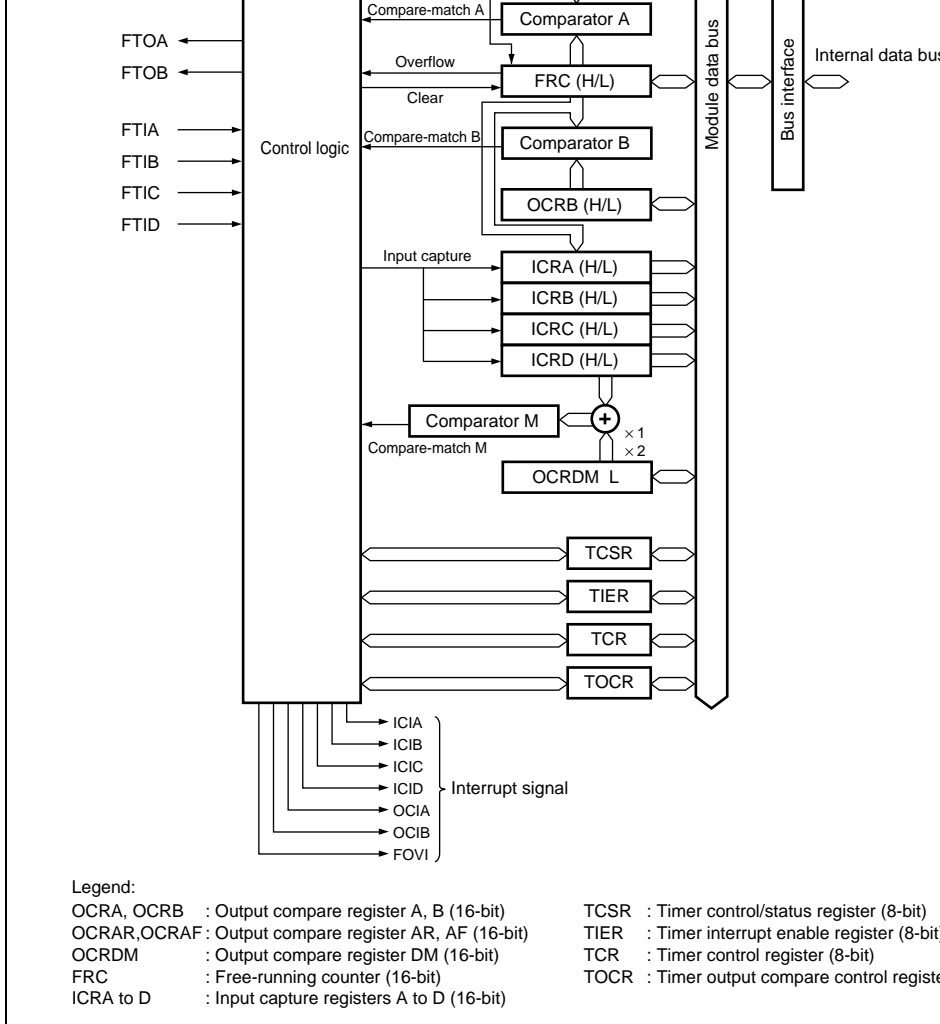


Figure 11.1 Block Diagram of 16-Bit Free-Running Timer

Output compare B output pin	FTOB	Output	Output compare B output
Input capture A input pin	FTIA	Input	Input capture A input
Input capture B input pin	FTIB	Input	Input capture B input
Input capture C input pin	FTIC	Input	Input capture C input
Input capture D input pin	FTID	Input	Input capture D input

11.3 Register Descriptions

The FRT has the following registers.

- Free-running counter (FRC)
- Output compare register A (OCRA)
- Output compare register B (OCRB)
- Input capture register A (ICRA)
- Input capture register B (ICRB)
- Input capture register C (ICRC)
- Input capture register D (ICRD)
- Output compare register AR (OCRAR)
- Output compare register AF (OCRAF)
- Output compare register DM (OCRDM)
- Timer interrupt enable register (TIER)
- Timer control/status register (TCSR)
- Timer control register (TCR)
- Timer output compare control register (TOCR)

Note: OCRA and OCRB share the same address. Register selection is controlled by bit in TOCR. ICRA, ICRB, and ICRC share the same addresses with OCRAR and OCRDM. Register selection is controlled by the ICRS bit in TOCR.

The FRT has two output compare registers, OCRA and OCRB, each of which is a 16-bit readable/writable register whose contents are continually compared with the value in FRC. When a match is detected (compare-match), the corresponding output compare flag (OCFA or OCFB) is set to 1 in TCSR. If the OEA or OEB bit in TOCR is set to 1, when the OCR and FRC match, the output level selected by the OLVLA or OLVLB bit in TOCR is output at the compare output pin (FTOA or FTOB). Following a reset, the FTOA and FTOB output is high until the first compare-match. OCR should always be accessed in 16-bit units; cannot be accessed in 8-bit units. OCR is initialized to H'FFFF.

11.3.3 Input Capture Registers A to D (ICRA to ICRD)

The FRT has four input capture registers, ICRA to ICRD, each of which is a 16-bit readable/writable register. When the rising or falling edge of the signal at an input capture input pin (FTIA to FTID) is detected, the current FRC value is transferred to the corresponding input capture register (ICRA to ICRD). At the same time, the corresponding input capture flag (ICFA to ICFD) in TCSR is set to 1. The FRC contents are transferred to ICR regardless of the value of ICF. The input capture edge is selected by the input edge select bits (IEDGA to IEDGD) in TCR.

ICRC and ICRD can be used as ICRA and ICRB buffer registers, respectively, by means of the enable bits A and B (BUFEA and BUFEB) in TCR. For example, if an input capture occurs on FTIA and ICRC is specified as the ICRA buffer register, the FRC contents are transferred to ICR. If BUFEA is set to 1, the FRC contents are transferred to the buffer register ICRC.

To ensure input capture, the input capture pulse width should be at least 1.5 system clocks on a single edge. When triggering is enabled on both edges, the input capture pulse width should be at least 2.5 system clocks (ϕ).

ICRA to ICRD should always be accessed in 16-bit units; cannot be accessed in 8-bit units. ICRA to ICRD are initialized to H'0000.

OCRAF. The value of the OCVLA bit in TOCR is ignored, and 1 is output on a compare-match A following addition of OCRAF, while 0 is output on a compare-match A following addition of OCRAR.

When using the OCRA automatic addition function, do not select internal clock $\phi/2$ as input clock together with a set value of H'0001 or less for OCRAR (or OCRAF).

OCRAR and OCRAF should always be accessed in 16-bit units; cannot be accessed in 8-bit units. OCRAR and OCRAF are initialized to H'FFFF.

11.3.5 Output Compare Register DM (OCRDM)

OCRDM is a 16-bit readable/writable register in which the upper 8 bits are fixed at H'0000. When the ICRDMS bit in TOCR is set to 1 and the contents of OCRDM are other than H'0000, the operation of ICRD is changed to include the use of OCRDM. The point at which input capture D occurs is taken as the start of a mask interval. Next, twice the contents of OCRDM is added to the contents of ICRD, and the result is compared with the FRC value. The point at which a match is taken as the end of the mask interval. New input capture D events are disabled during the mask interval. A mask interval is not generated when the contents of OCRDM are H'0000. The ICRDMS bit is set to 1.

OCRDM should always be accessed in 16-bit units; cannot be accessed in 8-bit units. OCRDM is initialized to H'0000.

				0: ICIA requested by ICFA is disabled 1: ICIA requested by ICFA is enabled
6	ICIBE	0	R/W	Input Capture Interrupt B Enable Selects whether to enable input capture interrupt request (ICIB) when input capture flag B (ICIFB) is set to 1. 0: ICIB requested by ICFB is disabled 1: ICIB requested by ICFB is enabled
5	ICICE	0	R/W	Input Capture Interrupt C Enable Selects whether to enable input capture interrupt request (ICIC) when input capture flag C (ICIFC) is set to 1. 0: ICIC requested by ICFC is disabled 1: ICIC requested by ICFC is enabled
4	ICIDE	0	R/W	Input Capture Interrupt D Enable Selects whether to enable input capture interrupt request (ICID) when input capture flag D (ICIFD) is set to 1. 0: ICID requested by ICFD is disabled 1: ICID requested by ICFD is enabled
3	OCIAE	0	R/W	Output Compare Interrupt A Enable Selects whether to enable output compare interrupt request (OCIA) when output compare flag A (OCIFA) is set to 1. 0: OCIA requested by OCFA is disabled 1: OCIA requested by OCFA is enabled

Selects whether to enable a free-running timer interrupt (FOVI) when the timer overflow flag (OVF) in TCSR is set to 1.

0: FOVI requested by OVF is disabled

1: FOVI requested by OVF is enabled

0	—	1	R	Reserved
---	---	---	---	----------

This bit is always read as 1 and cannot be

11.3.7 Timer Control/Status Register (TCSR)

TCSR is used for counter clear selection and control of interrupt request signals.

Bit	Bit Name	Initial Value	R/W	Description
7	ICFA	0	R/(W)*	Input Capture Flag A
<p>This status flag indicates that the FRC value has been transferred to ICRA by means of an input capture signal. When BUFEA = 1, ICFA indicates that the ICRA value has been moved into ICRC and the FRC value has been transferred to ICRA. Clearing this bit written to this bit to clear the flag.</p> <p>[Setting condition]</p> <p>When an input capture signal causes the FRC value to be transferred to ICRA</p> <p>[Clearing condition]</p> <p>Read ICFA when ICFA = 1, then write 0 to</p>				

[Setting condition]
When an input capture signal causes the FRC value to be transferred to ICRB

[Clearing condition]

Read ICFB when ICFB = 1, then write 0 to ICFB

5 ICFC 0

R/(W)* Input Capture Flag C

This status flag indicates that the FRC value has been transferred to ICRC by means of an input capture signal. When BUFEA = 1, on occurrence of a capture signal specified by the IEDGC bit at the input pin, ICFC is set but data is not transferred to ICRC. In buffer operation, ICFC can be used as an external interrupt signal by setting the ICICE bit. Only 0 can be written to this bit to clear the flag.

[Setting condition]

When an input capture signal is received

[Clearing condition]

Read ICFC when ICFC = 1, then write 0 to ICFC

4 ICFD 0

R/(W)* Input Capture Flag D

This status flag indicates that the FRC value has been transferred to ICRD by means of an input capture signal. When BUFEA = 1, on occurrence of a capture signal specified by the IEDGD bit at the input pin, ICFD is set but data is not transferred to ICRD. In buffer operation, ICFD can be used as an external interrupt signal by setting the ICIDE bit. Only 0 can be written to this bit to clear the flag.

[Setting condition]

When an input capture signal is received

[Clearing condition]

Read ICFD when ICFD = 1, then write 0 to ICFD

				Read OCFA when OCFA = 1, then write 0 to clear the flag.
2	OCFB	0	R/(W)*	<p>Output Compare Flag B</p> <p>This status flag indicates that the FRC value is equal to the OCRB value. Only 0 can be written to this bit to clear the flag.</p> <p>[Setting condition] When FRC = OCRB</p> <p>[Clearing condition] Read OCFB when OCFB = 1, then write 0 to clear the flag.</p>
1	OVF	0	R/(W)*	<p>Timer Overflow</p> <p>This status flag indicates that the FRC has overflowed. Only 0 can be written to this bit to clear the flag.</p> <p>[Setting condition] When FRC overflows (changes from H'FFFF to H'0000)</p> <p>[Clearing condition] Read OVF when OVF = 1, then write 0 to clear the flag.</p>
0	CCLRA	0	R/W	<p>Counter Clear A</p> <p>This bit selects whether the FRC is to be cleared at compare-match A (when the FRC and OCRB match).</p> <p>0: FRC clearing is disabled 1: FRC is cleared at compare-match A</p>

Note: * Only 0 can be written to clear the flag.

				signal (FTIA). 0: Capture on the falling edge of FTIA 1: Capture on the rising edge of FTIA
6	IEDGB	0	R/W	Input Edge Select B Selects the rising or falling edge of the input signal (FTIB). 0: Capture on the falling edge of FTIB 1: Capture on the rising edge of FTIB
5	IEDGC	0	R/W	Input Edge Select C Selects the rising or falling edge of the input signal (FTIC). 0: Capture on the falling edge of FTIC 1: Capture on the rising edge of FTIC
4	IEDGD	0	R/W	Input Edge Select D Selects the rising or falling edge of the input signal (FTID). 0: Capture on the falling edge of FTID 1: Capture on the rising edge of FTID
3	BUFEA	0	R/W	Buffer Enable A Selects whether ICRC is to be used as a buffer register for ICRA. 0: ICRC is not used as a buffer register for ICRA 1: ICRC is used as a buffer register for ICRA
2	BUFEB	0	R/W	Buffer Enable B Selects whether ICRD is to be used as a buffer register for ICRB. 0: ICRD is not used as a buffer register for ICRB 1: ICRD is used as a buffer register for ICRB

11.3.9 Timer Output Compare Control Register (TOCR)

TOCR enables output from the output compare pins, selects the output levels, switches between output compare registers A and B, controls the ICRD and OCRA operating modes, and switches access to input capture registers A, B, and C.

Bit	Bit Name	Initial Value	R/W	Description
7	ICRDMS	0	R/W	Input Capture D Mode Select Specifies whether ICRD is used in the normal operating mode or in the operating mode using OCRDM. 0: The normal operating mode is specified for ICRD. 1: The operating mode using OCRDM is specified for ICRD.
6	OCRAMS	0	R/W	Output Compare A Mode Select Specifies whether OCRA is used in the normal operating mode or in the operating mode using OCRAR and OCRAF. 0: The normal operating mode is specified for OCRA. 1: The operating mode using OCRAR and OCRAF is specified for OCRA.
5	ICRS	0	R/W	Input Capture Register Select The same addresses are shared by ICRA and OCRA, by ICRB and OCRAF, and by ICRC and OCRA. The ICRS bit determines which registers are selected. When the shared addresses are read from or written to, the operation of ICRA, ICRB, and ICRC is not affected. 0: ICRA, ICRB, and ICRC are selected. 1: OCRAR, OCRAF, and OCRDM are selected.

3	OEA	0	R/W	<p>Output Enable A</p> <p>Enables or disables output of the output compare output pin (FTOA).</p> <p>0: Output compare A output is disabled</p> <p>1: Output compare A output is enabled</p>
2	OEB	0	R/W	<p>Output Enable B</p> <p>Enables or disables output of the output compare output pin (FTOB).</p> <p>0: Output compare B output is disabled</p> <p>1: Output compare B output is enabled</p>
1	OLVLA	0	R/W	<p>Output Level A</p> <p>Selects the level to be output at the output compare output pin (FTOA) in response to compare-match (signal indicating a match between the FRC values). When the OCRAMS bit is 1, this bit is ignored.</p> <p>0: 0 is output at compare-match A</p> <p>1: 1 is output at compare-match A</p>
0	OLVLB	0	R/W	<p>Output Level B</p> <p>Selects the level to be output at the output compare output pin (FTOB) in response to compare-match (signal indicating a match between the FRC values).</p> <p>0: 0 is output at compare-match B</p> <p>1: 1 is output at compare-match B</p>

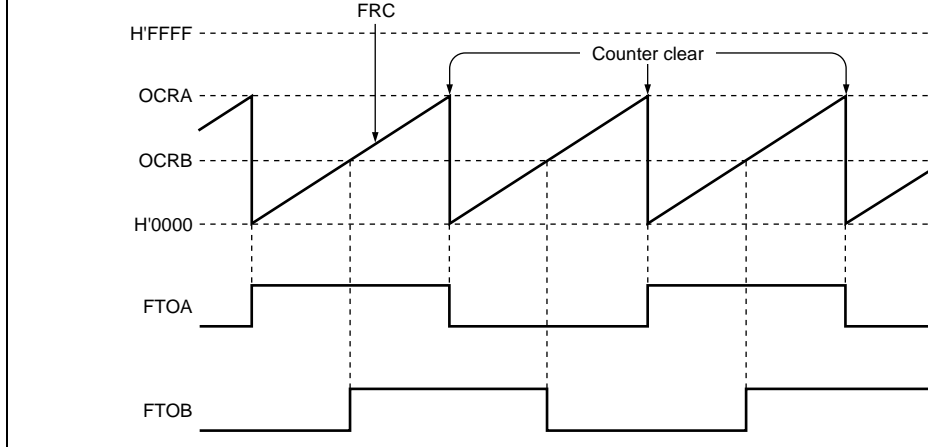


Figure 11.2 Example of Pulse Output

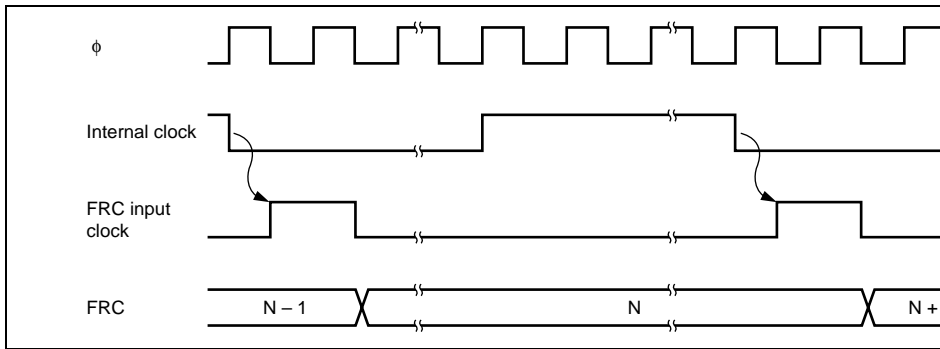


Figure 11.3 Increment Timing with Internal Clock Source

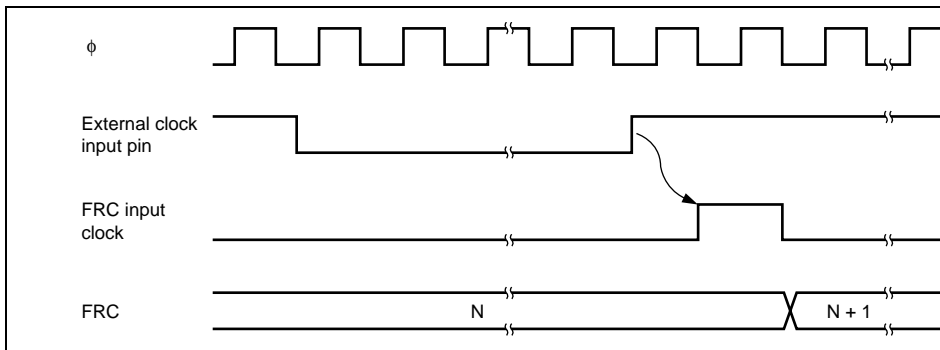


Figure 11.4 Increment Timing with External Clock Source

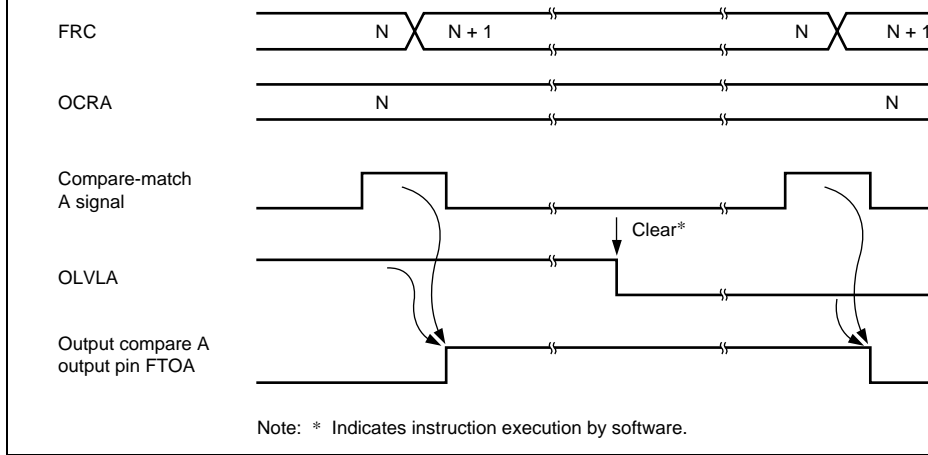


Figure 11.5 Timing of Output Compare A Output

11.5.3 FRC Clear Timing

FRC can be cleared when compare-match A occurs. Figure 11.6 shows the timing of this operation.

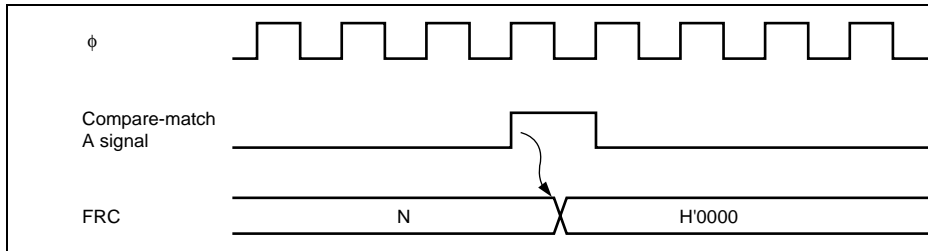


Figure 11.6 Clearing of FRC by Compare-Match A Signal

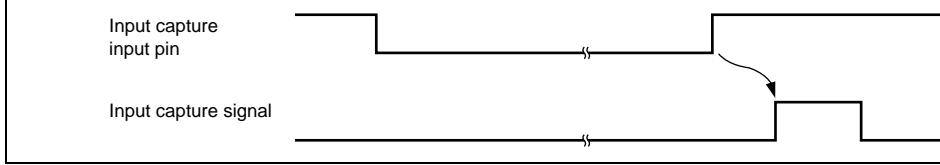


Figure 11.7 Input Capture Input Signal Timing (Usual Case)

If ICRA to ICRAD are read when the corresponding input capture signal arrives, the input capture signal is delayed by one system clock (ϕ). Figure 11.8 shows the timing for this case.

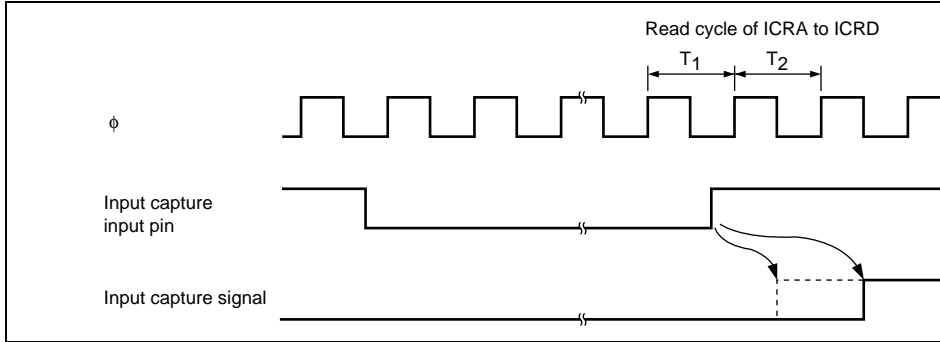


Figure 11.8 Input Capture Input Signal Timing (When ICRA to ICRD are Read)

11.5.5 Buffered Input Capture Input Timing

ICRC and ICRD can operate as buffers for ICRA and ICRB, respectively. Figure 11.9 shows input capture operates when ICRC is used as ICRA's buffer register (BUFEA = 1) and IEDGA and IEDGC are set to different values (IEDGA = 0 and IEDGC = 1, or IEDGA = 1 and IEDGC = 0) so that input capture is performed on both the rising and falling edges of FTIA.

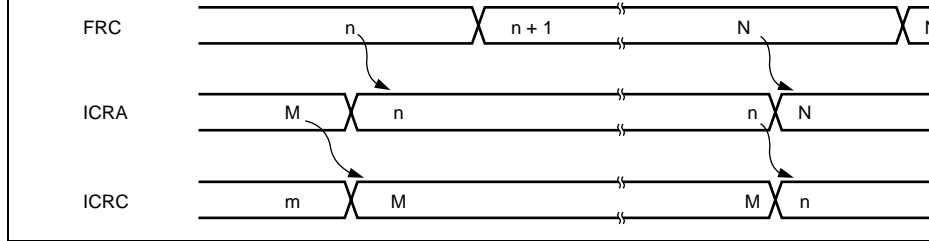


Figure 11.9 Buffered Input Capture Timing

Even when ICRC or ICRD is used as a buffer register, its input capture flag is set by the transition of its input capture signal. For example, if ICRC is used to buffer ICRA, when the transition selected by the IEDGC bit occurs on the FTIC input capture line, ICFC will be set. If the ICICE bit is set at this time, an interrupt will be requested. The FRC value will not be transferred to ICRC, however. In buffered input capture, if either set of two registers to be transferred (ICRA and ICRC, or ICRB and ICRD) is being read when the input capture signal arrives, input capture is delayed by one system clock (ϕ). Figure 11.10 shows the timing when BUFEA = 1.

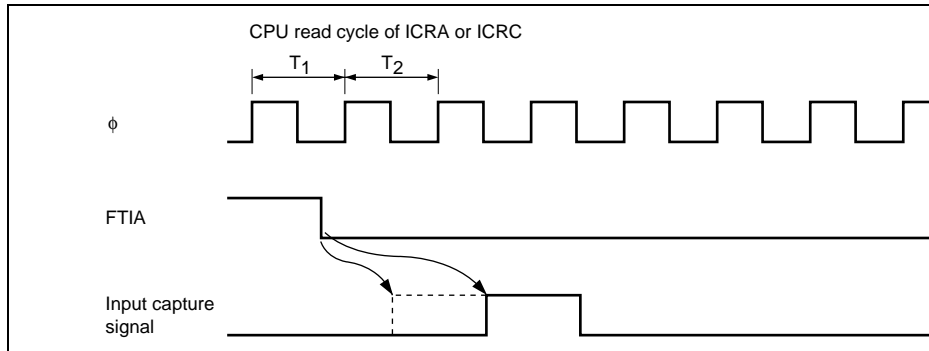


Figure 11.10 Buffered Input Capture Timing (BUFEA = 1)

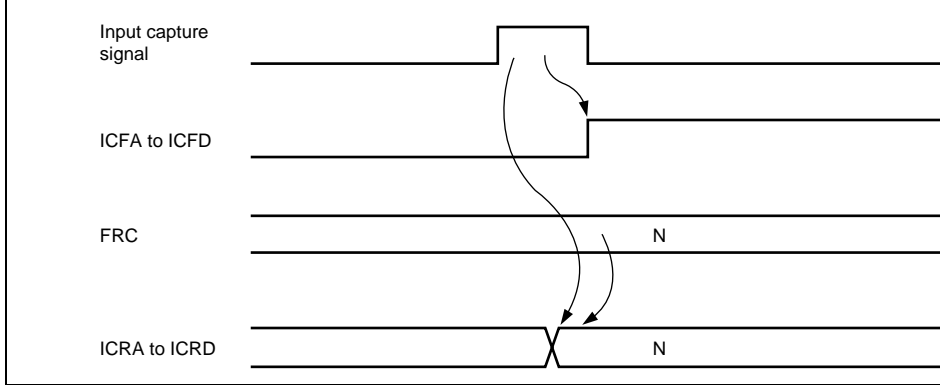


Figure 11.11 Timing of Input Capture Flag (ICFA, ICFB, ICFC, or ICFD) S

11.5.7 Timing of Output Compare Flag (OCF) setting

The output compare flag, OCFA or OCFB, is set to 1 by a compare-match signal generated when the FRC value matches the OCRA or OCRB value. This compare-match signal is generated at the last state in which the two values match, just before FRC increments to a new value. When the FRC and OCRA or OCRB value match, the compare-match signal is not generated until the next cycle of the clock source. Figure 11.12 shows the timing of setting the OCFA or OCFB.

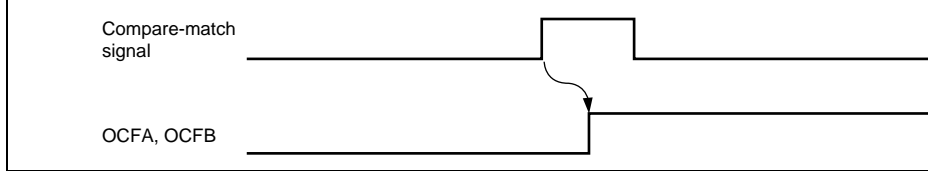


Figure 11.12 Timing of Output Compare Flag (OCFA or OCFB) Setting

11.5.8 Timing of FRC Overflow Flag Setting

The FRC overflow flag (OVF) is set to 1 when FRC overflows (changes from H'FFFF to H'0000). Figure 11.13 shows the timing of setting the OVF flag.

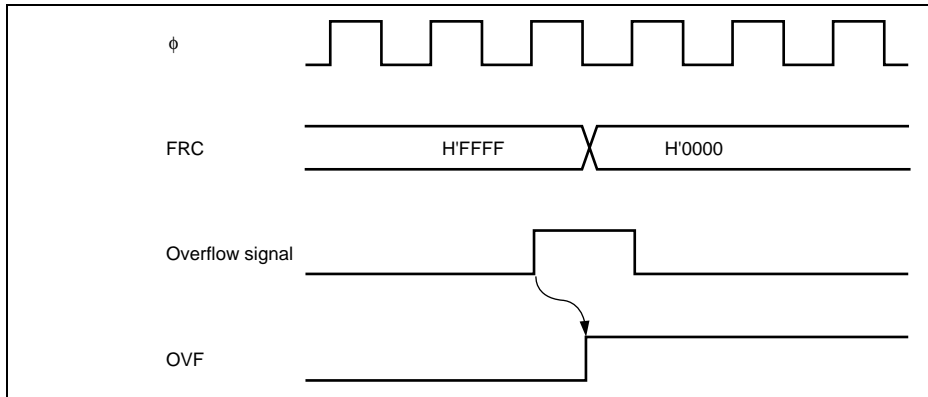


Figure 11.13 Timing of Overflow Flag (OVF) Setting

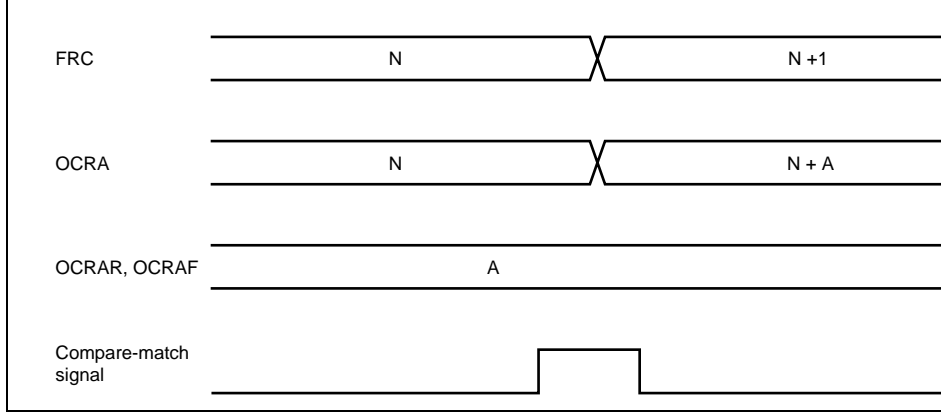


Figure 11.14 OCRA Automatic Addition Timing

11.5.10 Mask Signal Generation Timing

When the ICRDMS bit in TOCR is set to 1 and the contents of OCRDM are other than 0, a mask signal that masks the ICRD input capture signal is generated. The mask signal is set by the sum of the ICRD contents and twice the OCRDM contents, and an FRC compare-match signal. Figure 11.15 shows the timing of setting the mask signal. Figure 11.16 shows the timing of clearing the mask signal.

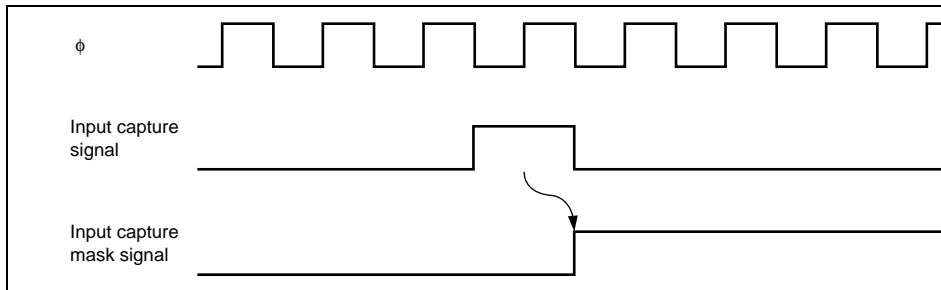


Figure 11.15 Timing of Input Capture Mask Signal Setting

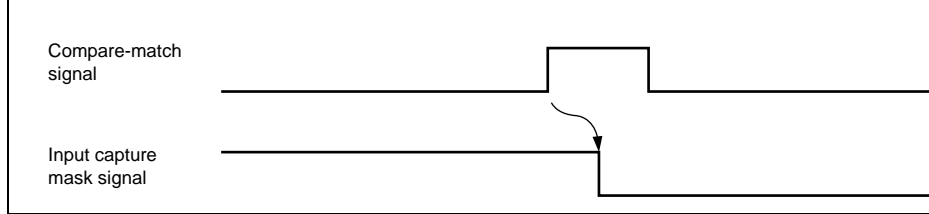


Figure 11.16 Timing of Input Capture Mask Signal Clearing

11.6 Interrupt Sources

The free-running timer can request seven interrupts: ICIA to ICID, OCIA, OCIB, and overflow interrupt. Each interrupt can be enabled or disabled by an enable bit in TIER. Independent signals are provided to the interrupt controller for each interrupt. Table 11.2 lists the sources and priorities of the interrupts.

The ICIA, ICIB, OCIA, and OCIB interrupts can be used as the on-chip DTC activation sources.

Table 11.2 FRT Interrupt Sources

Interrupt	Interrupt Source	Interrupt Flag	DTC Activation
ICIA	Input capture of ICRA	ICFA	Enabled
ICIB	Input capture of ICRB	ICFB	Enabled
ICIC	Input capture of ICRC	ICFC	Disabled
ICID	Input capture of ICRD	ICFD	Disabled
OCIA	Compare match of OCRA	OCFA	Enabled
OCIB	Compare match of OCRB	OCFB	Enabled
FOVI	Overflow of FRC	OVF	Disabled

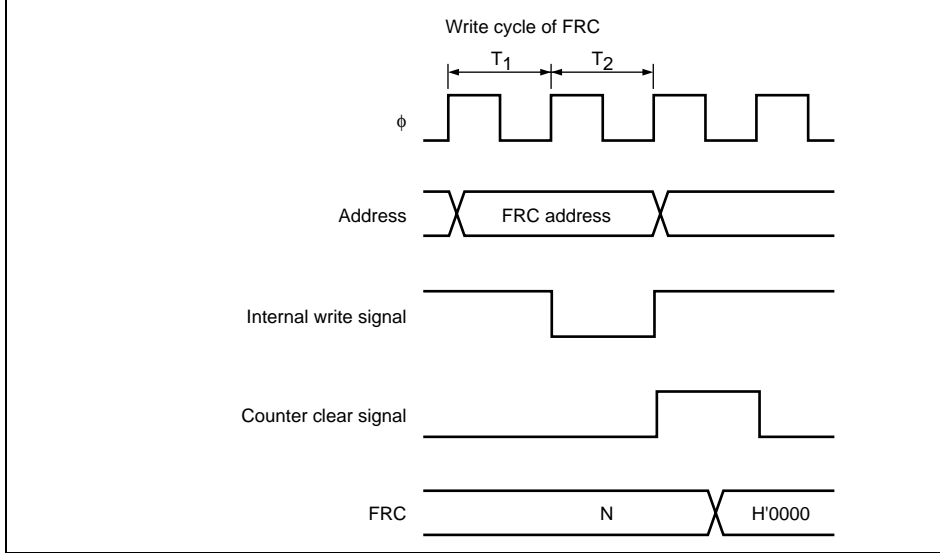


Figure 11.17 FRC Write-Clear Conflict

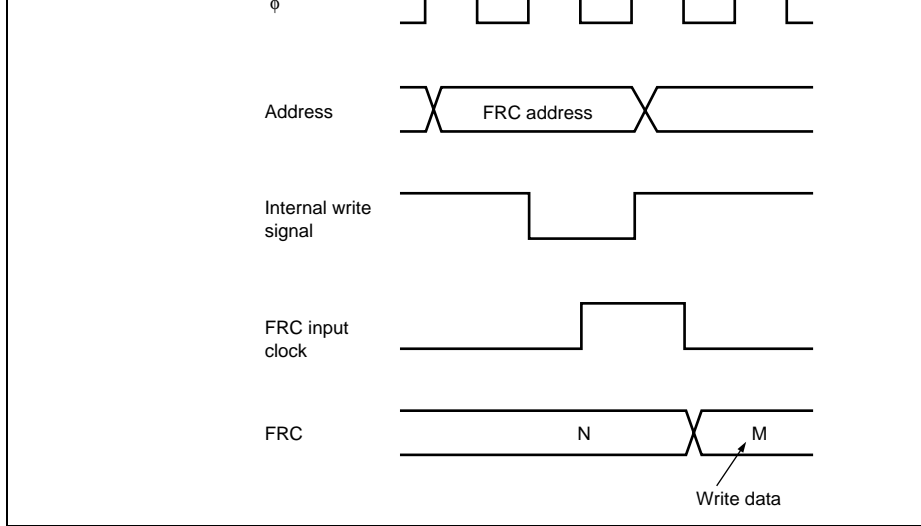


Figure 11.18 FRC Write-Increment Conflict

the automatic addition is not written to OCRA. Figure 11.20 shows the timing for this conflict.

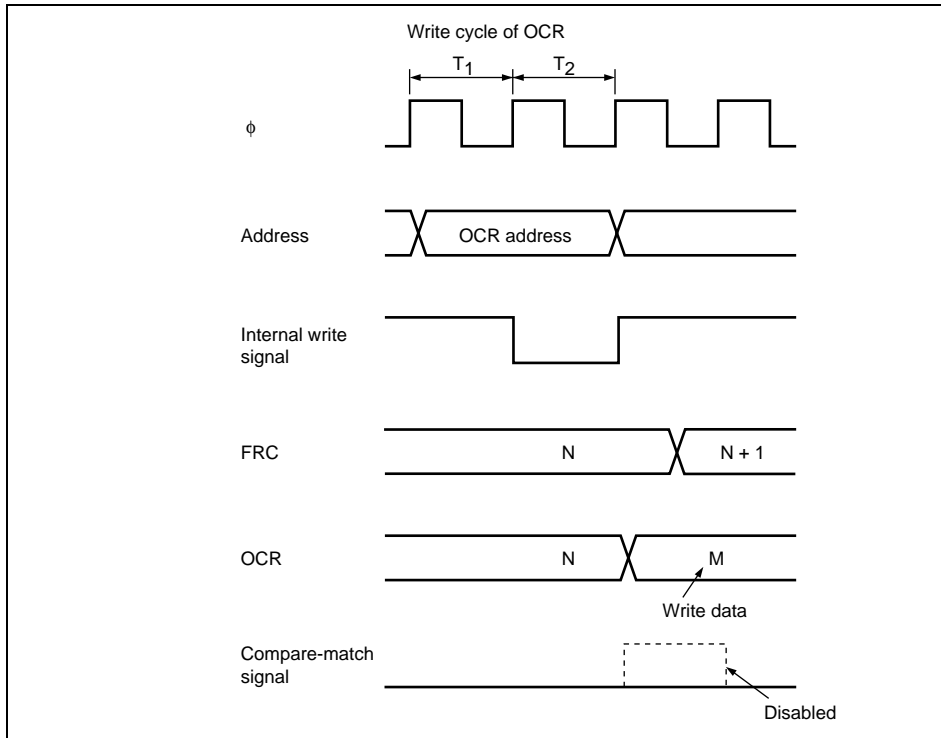


Figure 11.19 Conflict between OCR Write and Compare-Match (When Automatic Addition Function Is Not Used)

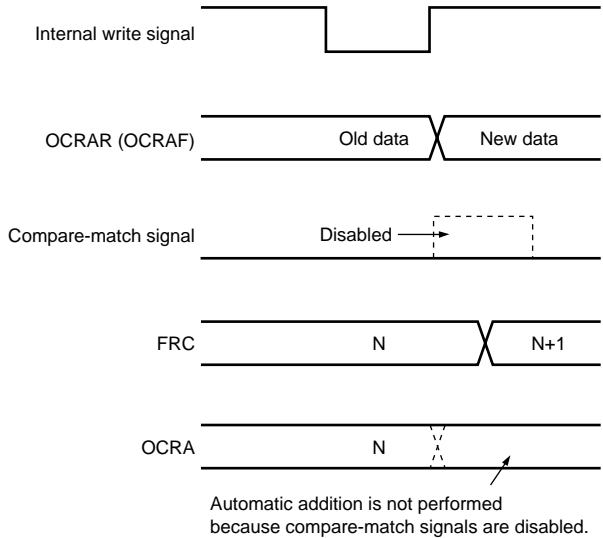
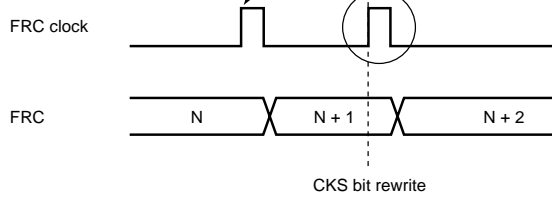


Figure 11.20 Conflict between OCRAR/OCRAF Write and Compare-M (When Automatic Addition Function Is Used)

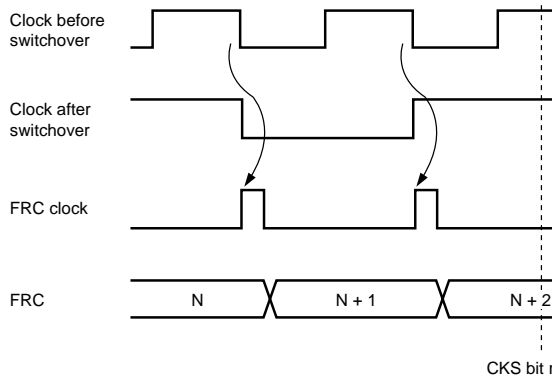
edge that triggers the FRC clock, and FRC is incremented. Switching between an internal and external clock can also cause FRC to increment.

Table 11.3 Switching of Internal Clock and FRC Operation

No.	Timing of Switchover by Means of CKS1 and CKS0 Bits	FRC Operation
1	Switching from low to low	<p data-bbox="864 627 982 643">CKS bit rewrite</p>
2	Switching from low to high	<p data-bbox="1039 1026 1157 1042">CKS bit rewrite</p>



4 Switching from high to high



Note: * Generated on the assumption that the switchover is a falling edge; FRC is

11.7.5 Module Stop Mode Setting

FRT operation can be enabled or disabled using the module stop control register. The setting is for FRT operation to be halted. Register access is enabled by canceling the mode. For details, refer to section 26, Power-Down Modes.



This LSI also has a similar on-chip 8-bit timer module (TMR_Y and TMR_X) with two channels, which can be used through connection to the timer connection.

12.1 Features

- Selection of clock sources
 - TMR_0, TMR_1: The counter input clock can be selected from six internal clocks or an external clock
 - TMR_Y, TMR_X: The counter input clock can be selected from three internal clocks or an external clock
- Selection of three ways to clear the counters
 - The counters can be cleared on compare-match A or compare-match B, or by a reset signal.
- Timer output controlled by two compare-match signals
 - The timer output signal in each channel is controlled by two independent compare-match signals, enabling the timer to be used for various applications, such as the general purpose pulse output or PWM output with an arbitrary duty cycle. (The TMR_Y does not have a timer output pin.)
- Cascading of TMR_0 and TMR_1
 - (TMR_Y and TMR_X cannot be cascaded.)
 - Operation as a 16-bit timer can be performed using TMR_0 as the upper half and TMR_1 as the lower half (16-bit count mode).
 - TMR_1 can be used to count TMR_0 compare-match occurrences (compare-match mode).
- Multiple interrupt sources for each channel
 - TMR_0, TMR_1, and TMR_Y: Three types of interrupts: Compare-match A, compare-match B, and overflow
 - TMR_X: Input capture

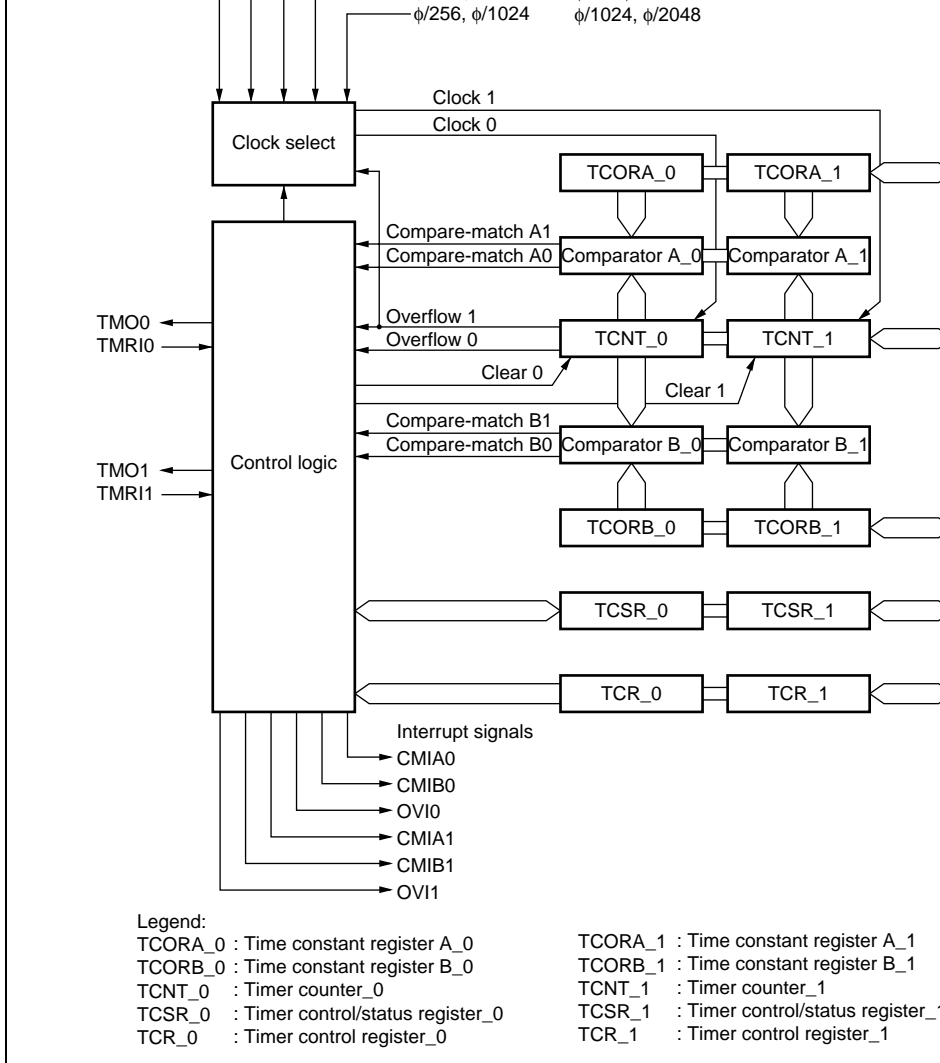


Figure 12.1 Block Diagram of 8-Bit Timers (TMR_0 and TMR_1)

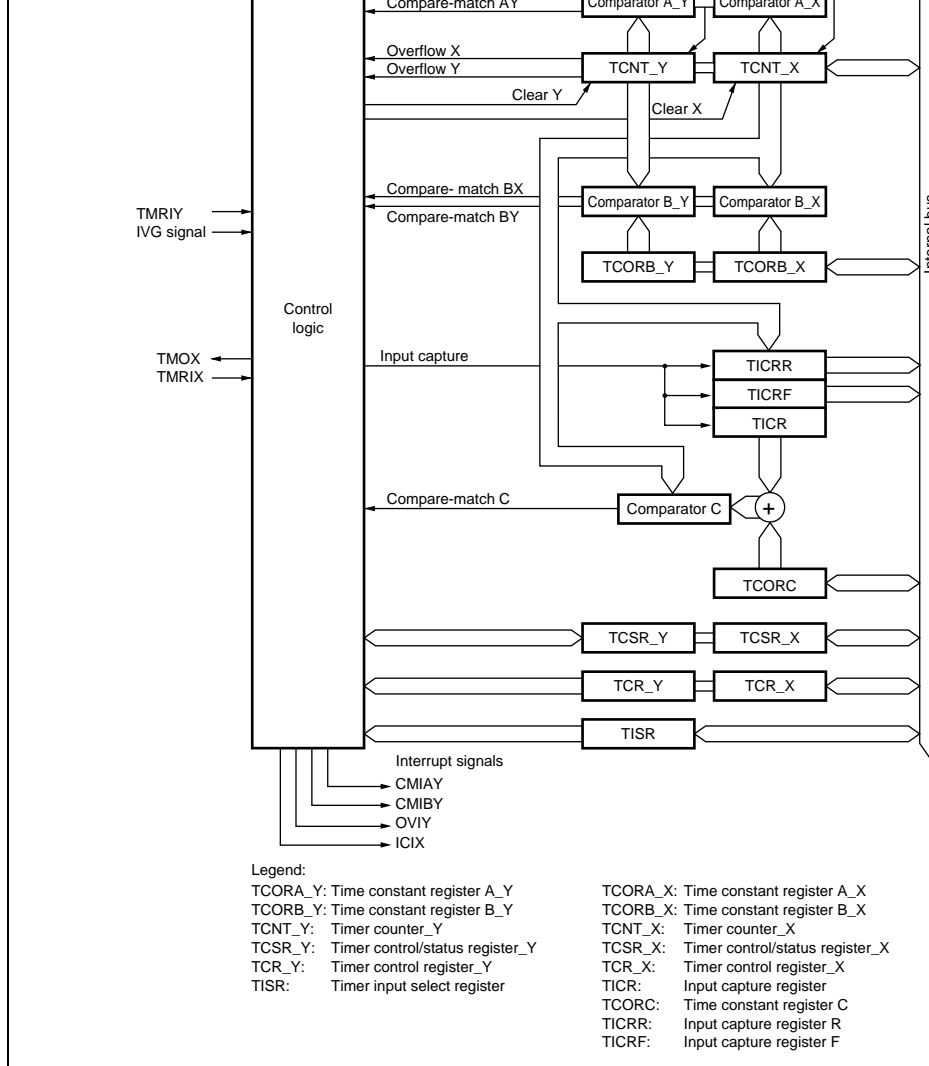


Figure 12.2 Block Diagram of 8-Bit Timers (TMR_Y and TMR_X)

	Timer reset input	TMRI0	Input	External reset input for the counter
TMR_1	Timer output	TMO1	Output	Output controlled by compare mode
	Timer clock input	TMCI1	Input	External clock input for the counter
	Timer reset input	TMRI1	Input	External reset input for the counter
TMR_Y	Timer clock/reset input	VSYNCl/TMIY (TMCiY/TMRIY)	Input	External clock input/external reset input for the counter
TMR_X	Timer output	TMOX	Output	Output controlled by compare mode
	Timer clock/reset input	HFBACkI/TMIx (TMCIX/TMRIx)	Input	External clock input/external reset input for the counter

12.3 Register Descriptions

The TMR has the following registers. For details on the serial timer control register, refer to section 3.2.3, Serial Timer Control Register (STCR). For details on timer connection register, refer to section 13.3.3, Timer Connection Register S (TCONRS).

- Timer counter (TCNT)
- Time constant register A (TCORA)
- Time constant register B (TCORB)
- Timer control register (TCR)
- Timer control/status register (TCSR)
- Timer input select register (TISR)^{*1}
- Time constant register C (TCORC)^{*2}
- Input capture register R (TICRR)^{*2}
- Input capture register F (TICRF)^{*2}

Notes: 1. TISR is only for the TMR_Y.

2. TCORC, TICRR, and TICRF are only for the TMR_X.

12.3.2 Time Constant Register A (TCORA)

TCORA is an 8-bit readable/writable register. TCORA_0 and TCORA_1 comprise a 16-bit register, so they can be accessed together by word access. TCORA is continually compared to the value in TCNT. When a match is detected, the corresponding compare-match flag CMFA in TCSR is set to 1. Note however that comparison is disabled during the T2 state of a write cycle. The timer output from the TMO pin can be freely controlled by these compare signals and the settings of output select bits OS1 and OS0 in TCSR. TCORA is initialized to 0xFF.

12.3.3 Time Constant Register B (TCORB)

TCORB is an 8-bit readable/writable register. TCORB_0 and TCORB_1 comprise a 16-bit register, so they can be accessed together by word access. TCORB is continually compared to the value in TCNT. When a match is detected, the corresponding compare-match flag CMFB in TCSR is set to 1. Note however that comparison is disabled during the T2 state of a write cycle. The timer output from the TMO pin can be freely controlled by these compare signals and the settings of output select bits OS3 and OS2 in TCSR. TCORB is initialized to 0xFF.

12.3.4 Timer Control Register (TCR)

TCR selects the TCNT clock source and the condition by which TCNT is cleared, and enables/disables interrupt requests.

6	CMIEA	0	R/W	Compare-Match Interrupt Enable A Selects whether the CMFA interrupt request (CMFA) is enabled or disabled when the CMFA flag in TCSR is set to 1. Note that a CMIA interrupt is not generated by TCSR regardless of the CMIEA value. 0: CMFA interrupt request (CMIA) is disabled 1: CMFA interrupt request (CMIA) is enabled
5	OVIE	0	R/W	Timer Overflow Interrupt Enable Selects whether the OVF interrupt request (OVF) is enabled or disabled when the OVF flag in TCSR is set to 1. Note that an OVI interrupt is not generated by TMR_X if OVF is disabled regardless of the OVIE value. 0: OVF interrupt request (OVI) is disabled 1: OVF interrupt request (OVI) is enabled
4	CCLR1	0	R/W	Counter Clear 1, 0
3	CCLR0	0	R/W	These bits select the method by which the timer is cleared. 00: Clearing is disabled 01: Cleared on compare-match A 10: Cleared on compare-match B 11: Cleared on rising edge of external reset input
2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	These bits select the clock input to TCNT and counter condition, together with the ICKS1 and ICKS0 bits.
0	CKS0	0	R/W	For details, see table 12.2.

	0	1	0	—	1	Increments at falling edge of internal clock
	0	1	1	—	0	Increments at falling edge of internal clock
	0	1	1	—	1	Increments at falling edge of internal clock
	1	0	0	—	—	Increments at overflow signal from TCNT_0
TMR_1	0	0	0	—	—	Disables clock input
	0	0	1	0	—	Increments at falling edge of internal clock
	0	0	1	1	—	Increments at falling edge of internal clock
	0	1	0	0	—	Increments at falling edge of internal clock
	0	1	0	1	—	Increments at falling edge of internal clock
	0	1	1	0	—	Increments at falling edge of internal clock
	0	1	1	1	—	Increments at falling edge of internal clock
	1	0	0	—	—	Increments at compare-match A from TCNT_0
TMR_Y	0	0	0	—	—	Disables clock input
	0	0	1	—	—	Increments at falling edge of internal clock
	0	1	0	—	—	Increments at falling edge of internal clock
	0	1	1	—	—	Increments at falling edge of internal clock
	1	0	0	—	—	Disables clock input
TMR_X	0	0	0	—	—	Disables clock input
	0	0	1	—	—	Increments at falling edge of internal clock
	0	1	0	—	—	Increments at falling edge of internal clock
	0	1	1	—	—	Increments at falling edge of internal clock
	1	0	0	—	—	Disables clock input
Common	1	0	1	—	—	Increments at rising edge of external clock
	1	1	0	—	—	Increments at falling edge of external clock
	1	1	1	—	—	Increments at both rising and falling edges of external clock.

Note: * If the TMR_0 clock input is set as the TCNT_1 overflow signal and the TMR_1 clock input is set as the TCNT_0 compare-match signal simultaneously, a count-up clock cannot be generated.

				[Setting condition] When the values of TCNT_0 and TCORB_0 n [Clearing conditions] <ul style="list-style-type: none"> • Read CMFB when CMFB = 1, then write 0 • When the DTC is activated by a CMIB inte
6	CMFA	0	R/(W)*	Compare-Match Flag A [Setting condition] When the values of TCNT_0 and TCORA_0 n [Clearing conditions] <ul style="list-style-type: none"> • Read CMFA when CMFA = 1, then write 0 • When the DTC is activated by a CMIA inte
5	OVF	0	R/(W)*	Timer Overflow Flag [Setting condition] When TCNT_0 overflows from H'FF to H'00 [Clearing condition] Read OVF when OVF = 1, then write 0 in OVF
4	ADTE	0	R/W	A/D Trigger Enable Enables or disables A/D converter start reques compare-match A. 0: A/D converter start requests by compare-m disabled 1: A/D converter start requests by compare-m enabled

				11: Output is inverted (toggle output)
1	OS1	0	R/W	Output Select 1, 0
0	OS0	0	R/W	These bits specify how the TMO0 pin output changed by compare-match A of TCORA_0 TCNT_0. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)

Note: * Only 0 can be written, for flag clearing.

- TCSR_1

Bit	Bit Name	Initial Value	R/W	Description
7	CMFB	0	R/(W)*	Compare-Match Flag B [Setting condition] When the values of TCNT_1 and TCORB_1 [Clearing conditions] <ul style="list-style-type: none"> • Read CMFB when CMFB = 1, then write • When the DTC is activated by a CMIB int
6	CMFA	0	R/(W)*	Compare-Match Flag A [Setting condition] When the values of TCNT_1 and TCORA_1 [Clearing conditions] <ul style="list-style-type: none"> • Read CMFA when CMFA = 1, then write • When the DTC is activated by a CMIA int

				This bit is always read as 1 and cannot be modified.
3	OS3	0	R/W	Output Select 3, 2
2	OS2	0	R/W	These bits specify how the TMO1 pin output level is changed by compare-match B of TCORB_1 and TCNT_1. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)
1	OS1	0	R/W	Output Select 1, 0
0	OS0	0	R/W	These bits specify how the TMO1 pin output level is changed by compare-match A of TCORA_1 and TCNT_1. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)

Note: * Only 0 can be written, for flag clearing.

				<ul style="list-style-type: none"> When the DTC is activated by a CMIB int
6	CMFA	0	R/(W)*1	<p>Compare-Match Flag A</p> <p>[Setting condition]</p> <p>When the values of TCNT_Y and TCORA_Y</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> Read CMFA when CMFA = 1, then write When the DTC is activated by a CMIA int
5	OVF	0	R/(W)*1	<p>Timer Overflow Flag</p> <p>[Setting condition]</p> <p>When TCNT_Y overflows from H'FF to H'00</p> <p>[Clearing condition]</p> <p>Read OVF when OVF = 1, then write 0 in OV</p>
4	ICIE	0	R/W	<p>Input Capture Interrupt Enable</p> <p>Enables or disables the ICF interrupt request the ICF bit in TCSR_X is set to 1.</p> <p>0: ICF interrupt request (ICIX) is disabled</p> <p>1: ICF interrupt request (ICIX) is enabled</p>

Bit	Bit Name	Initial Value	R/W	Description
1	OS1	0	R/W	Output Select 1, 0
0	OS0	0	R/W	These bits specify how the TMOY pin*2 output be changed by compare-match A of TCORA_ TCNT_Y. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)

- Notes: 1. Only 0 can be written, for flag clearing.
2. This product does not have a TMOY external output pin.

- TCSR_X

Bit	Bit Name	Initial Value	R/W	Description
7	CMFB	0	R/(W)*	Compare-Match Flag B [Setting condition] When the values of TCNT_X and TCORB_X r [Clearing conditions] <ul style="list-style-type: none"> • Read CMFB when CMFB = 1, then write 0 • When the DTC is activated by a CMIB inte
6	CMFA	0	R/(W)*	Compare-Match Flag A [Setting condition] When the values of TCNT_X and TCORA_X r [Clearing conditions] <ul style="list-style-type: none"> • Read CMFA when CMFA = 1, then write 0 • When the DTC is activated by a CMIA inte

[Setting condition]

When a rising edge and falling edge is detected, the external reset signal in that order, after the ICF (TCONRI) of the timer connection is set to 1

[Clearing condition]

Read ICF when ICF = 1, then write 0 in ICF

3	OS3	0	R/W	Output Select 3, 2
2	OS2	0	R/W	These bits specify how the TMOX pin output is changed by compare-match B of TCORB_X, TCNT_X. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)
1	OS1	0	R/W	Output Select 1, 0
0	OS0	0	R/W	These bits specify how the TMOX pin output is changed by compare-match A of TCORA_X, TCNT_X. 00: No change 01: 0 is output 10: 1 is output 11: Output is inverted (toggle output)

Note: * Only 0 can be written, for flag clearing.

TCORC is an 8-bit readable/writable register. The sum of contents of TCORC and TICNT is compared with TCNT. When a match is detected, a compare-match C signal is generated. However, comparison at the T2 state in the write cycle to TCORC and at the input capture T1 is disabled. TCORC is initialized to H'FF. The TCORC function is used for the timer connection. For details, refer to section 13, Timer Connection.

12.3.8 Input Capture Registers R and F (TICRR, TICRF)

TICRR and TICRF are 8-bit read-only registers. The contents of TCNT are transferred to TICRR on the rising edge and falling edge of the external reset input in that order, when the ICST bit of the timer connection is set to 1. The ICST bit is cleared to 0 when one capture operation is completed. TICRR and TICRF are initialized to H'00. The TICRR and TICRF functions are used for the timer connection. For details, refer to section 13, Timer Connection.

12.3.9 Timer Input Select Register (TISR)

TISR selects a signal source of external clock/reset input for the counter.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 1	R/(W)	Reserved The initial values should not be modified.
0	IS	0	R/W	Input Select Selects an internal synchronization signal (IVG) or timer clock/reset input pin VSYNCI/TMIY (TMCY/TMRIY) as the signal source of external clock/reset input for TMR_Y counter. 0: IVG signal is selected 1: VSYNCI/TMIY (TMCY/TMRIY) is selected

- Set the OS3 to OS0 bits in TCSR to B'0110 so that 1 is output according to the compare match of TCORA and 0 is output according to the compare match of TCORB.

According to the above settings, the waveforms with the TCORA cycle and TCORB p can be output without the intervention of software.

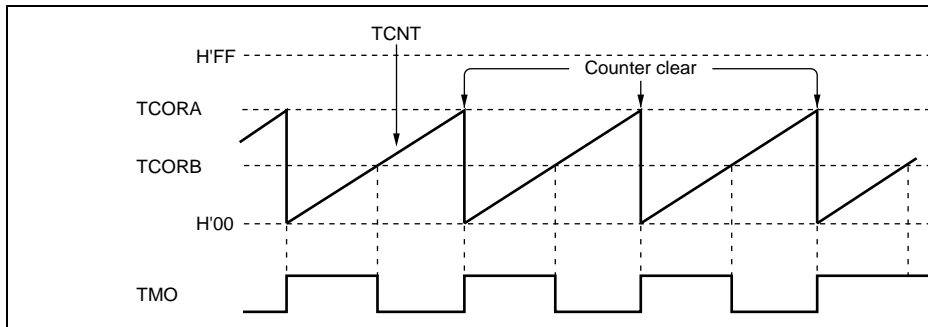


Figure 12.3 Pulse Output Example

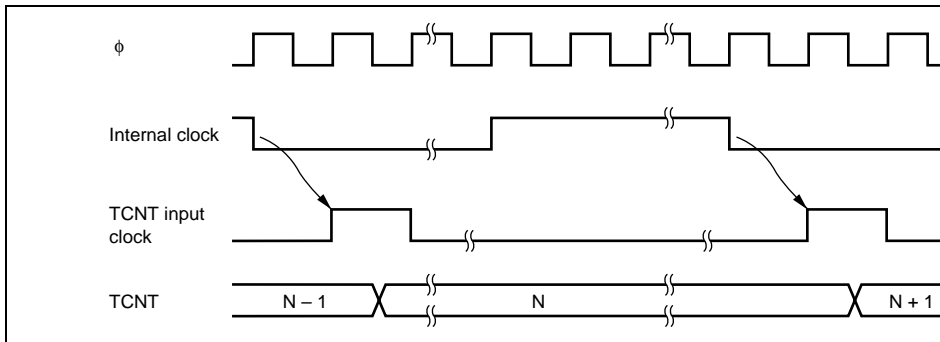


Figure 12.4 Count Timing for Internal Clock Input

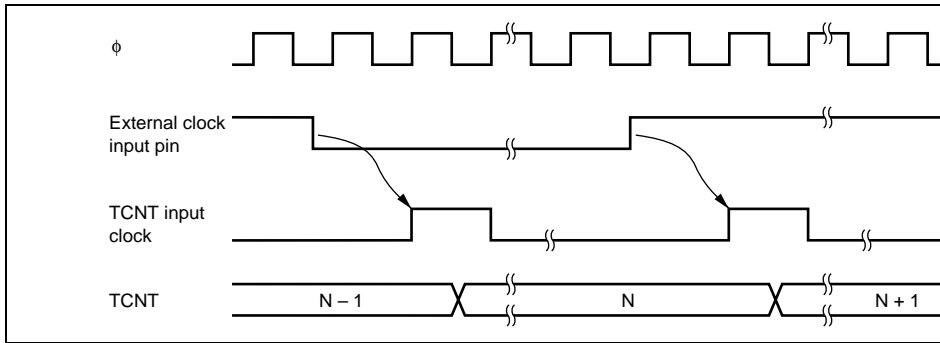


Figure 12.5 Count Timing for External Clock Input (Both Edges)

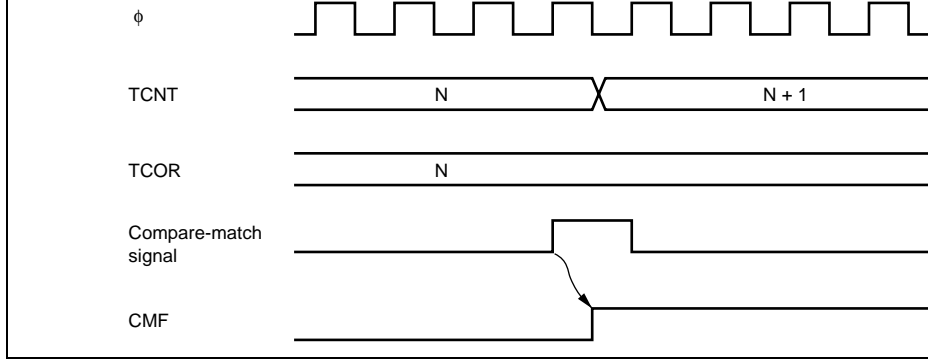


Figure 12.6 Timing of CMF Setting at Compare-Match

12.5.3 Timing of Timer Output at Compare-Match

When a compare-match signal occurs, the timer output changes as specified by the OS bits in TCSR. Figure 12.7 shows the timing of timer output when the output is set to toggle on compare-match A signal.

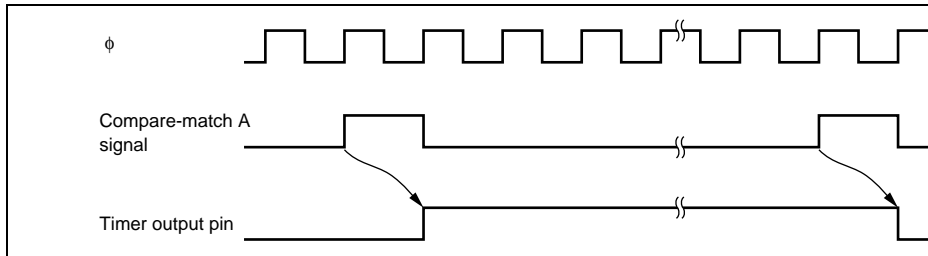


Figure 12.7 Timing of Toggled Timer Output by Compare-Match A Signal

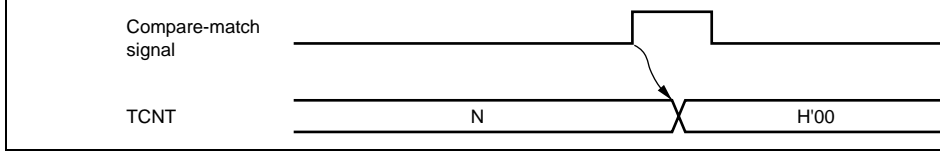


Figure 12.8 Timing of Counter Clear by Compare-Match

12.5.5 TCNT External Reset Timing

TCNT is cleared at the rising edge of an external reset input, depending on the settings CCLR1 and CCLR0 bits in TCR. The width of the clearing pulse must be at least 1.5 st. 12.9 shows the timing of clearing the counter by an external reset input.

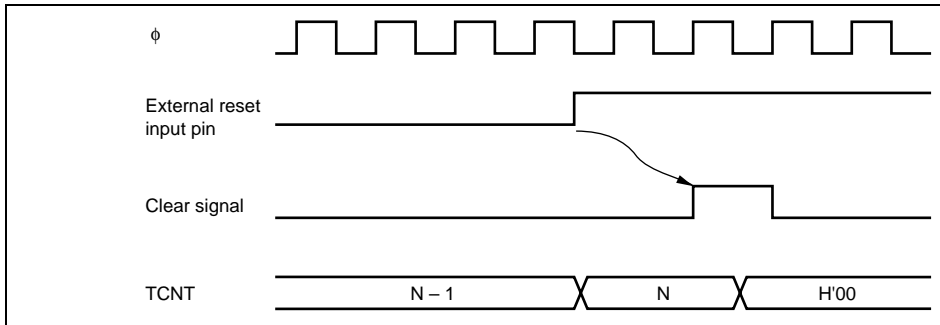


Figure 12.9 Timing of Counter Clear by External Reset Input

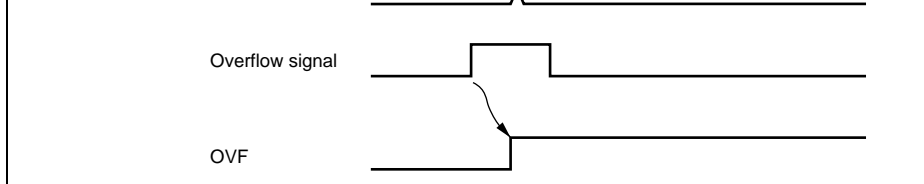


Figure 12.10 Timing of OVF Flag Setting

12.6 Operation with Cascaded Connection

If bits CKS2 to CKS0 in either TCR_0 or TCR_1 are set to B'100, the 8-bit timers of the two channels are cascaded. With this configuration, a single 16-bit timer can be used (16-bit count mode) or the compare-matches of the 8-bit timer of channel 0 can be counted by the 8-bit timer of channel 1 (compare-match count mode).

12.6.1 16-Bit Count Mode

When bits CKS2 to CKS0 in TCR_0 are set to B'100, the timer functions as a single 16-bit timer with channel 0 occupying the upper 8 bits and channel 1 occupying the lower 8 bits.

Setting of Compare-Match Flags:

- The CMF flag in TCSR_0 is set to 1 when a 16-bit compare-match occurs.
- The CMF flag in TCSR_1 is set to 1 when a lower 8-bit compare-match occurs.

Pin Output:

- Control of output from the TMO0 pin by bits OS3 to OS0 in TCSR_0 is in accordance with the 16-bit compare-match conditions.
- Control of output from the TMO1 pin by bits OS3 to OS0 in TCSR_1 is in accordance with the lower 8-bit compare-match conditions.

12.6.2 Compare-Match Count Mode

When bits CKS2 to CKS0 in TCR_1 are B'100, TCNT_1 counts the occurrence of compare-match conditions for channel 0. Channels 0 and 1 are controlled independently. Conditions such as setting the CMF flag, generation of interrupts, output from the TMO pin, and counter clearing are controlled in accordance with the settings for each channel.

12.7 Input Capture Operation

TMR_X has input capture registers (TICR, TICRR and TICRF). A narrow pulse width is measured with TICRR and TICRF, using a single capture operation controlled by the ICST bit in TCONRI of the timer connection. If the falling edge of TMRIX is detected after its rising edge has been detected while the ICST bit is set to 1, the value of TCNT at that time is transferred to TICRR and TICRF, and the ICST bit is cleared to 0.

The input signal to TMRIX can be switched by the setting of the other bits in TCONRI.

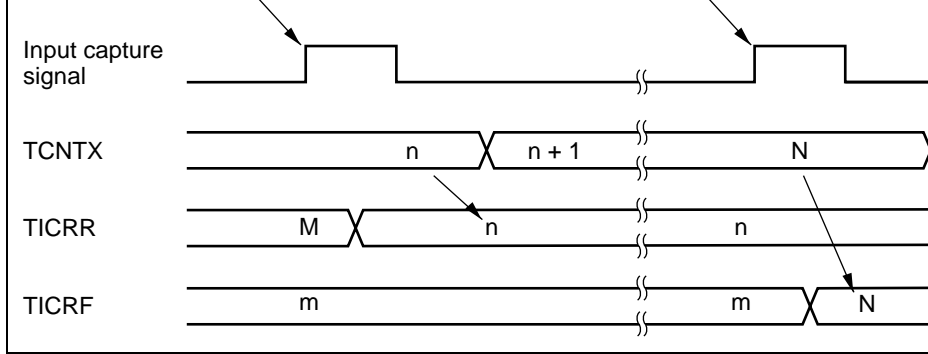
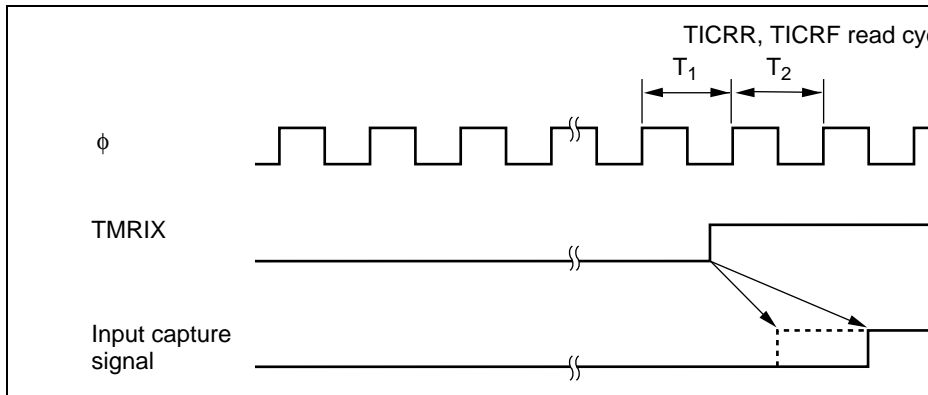


Figure 12.11 Timing of Input Capture Operation

If the input capture signal is input while TICRR and TICRF are being read, the input capture signal is delayed by one system clock (ϕ) cycle. Figure 12.12 shows the timing of this



**Figure 12.12 Timing of Input Capture Signal
(Input Capture Signal Is Input during TICRR and TICRF Read)**

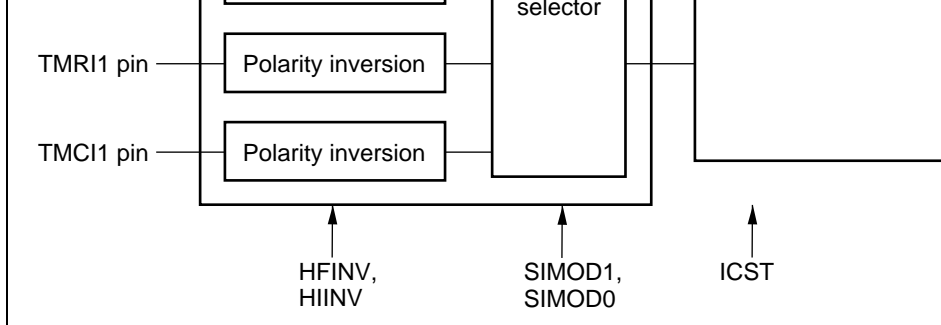


Figure 12.13 Input Capture Signal Selection

Table 12.3 Input Capture Signal Selection

TCONRI					Description	
Bit 4	Bit 7	Bit 6	Bit 3	Bit 1		
ICST	SIMOD1	SIMOD0	HFINV	HIINV		
0	—	—	—	—	Input capture function not used	
1	0	0	0	—	TMIX pin input selection	
			1	—	Inverted TMIX pin input selection	
		1	—	0	—	TMR11 pin input selection
			—	—	1	—
	1	1	—	0	—	TMC11 pin input selection
—	—	—	—	1	Inverted TMC11 pin input selection	

Table 12.4 Interrupt Sources of 8-Bit Timers TMR_0, TMR_1, TMR_Y, and TMR_X

Channel	Name	Interrupt Source	Interrupt Flag	DTC Activation	Interrupt Priority
TMR_0	CMIA0	TCORA_0 compare-match	CMFA	Enabled	High
	CMIB0	TCORB_0 compare-match	CMFB	Enabled	
	OVI0	TCNT_0 overflow	OVF	Disabled	
TMR_1	CMIA1	TCORA_1 compare-match	CMFA	Enabled	
	CMIB1	TCORB_1 compare-match	CMFB	Enabled	
	OVI1	TCNT_1 overflow	OVF	Disabled	
TMR_Y	CMIAY	TCORA_Y compare-match	CMFA	Enabled	
	CMIBY	TCORB_Y compare-match	CMFB	Enabled	
	OVIY	TCNT_Y overflow	OVF	Disabled	
TMR_X	ICIX	Input capture	ICF	Disabled	Low



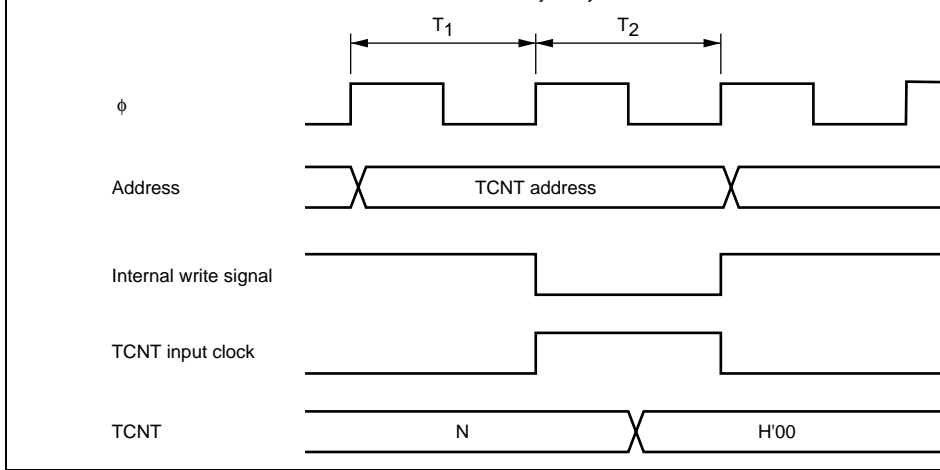


Figure 12.14 Conflict between TCNT Write and Clear

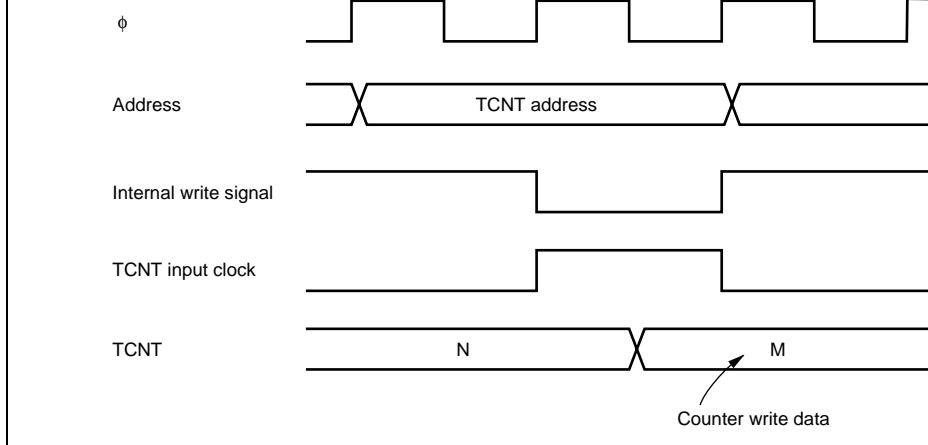


Figure 12.15 Conflict between TCNT Write and Increment

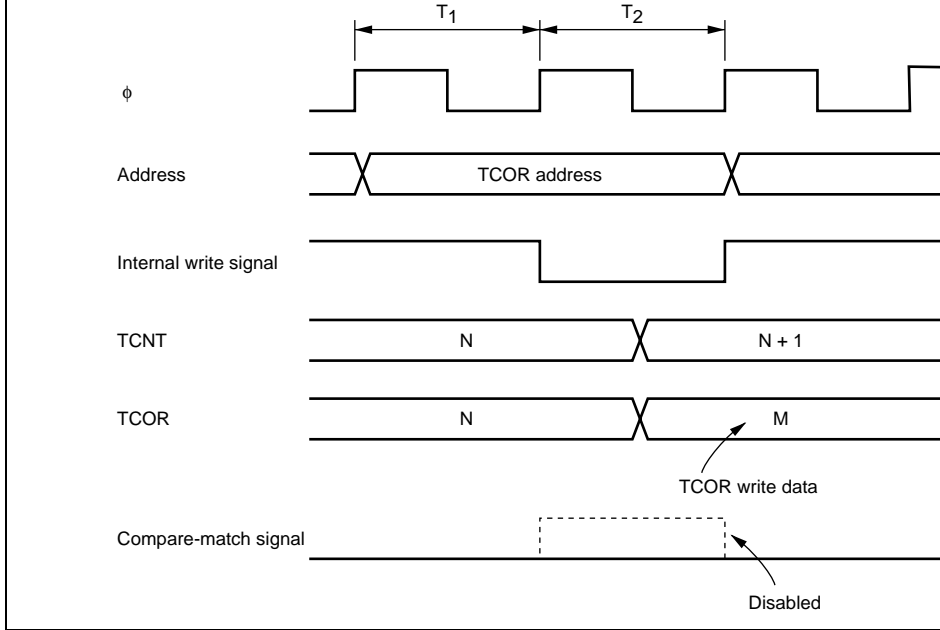


Figure 12.16 Conflict between TCOR Write and Compare-Match

Toggle output	High
1 output	↑
0 output	
No change	Low

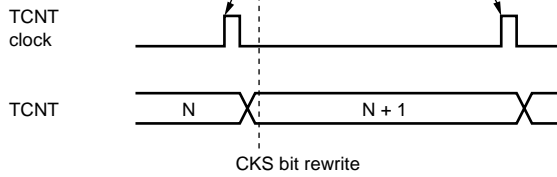
12.9.5 Switching of Internal Clocks and TCNT Operation

TCNT may increment erroneously when the internal clock is switched over. Table 12.6 shows the relationship between the timing at which the internal clock is switched (by writing to the CKS0 and CKS1 bits) and the TCNT operation.

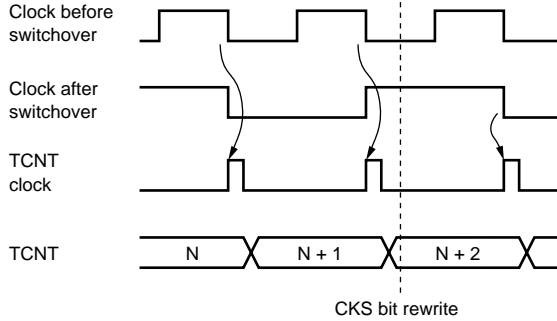
When the TCNT clock is generated from an internal clock, the falling edge of the internal clock pulse is detected. If clock switching causes a change from high to low level, as shown in table 12.6, a TCNT clock pulse is generated on the assumption that the switchover is a falling edge, and TCNT is incremented.

Erroneous incrementation can also happen when switching between internal and external clocks.

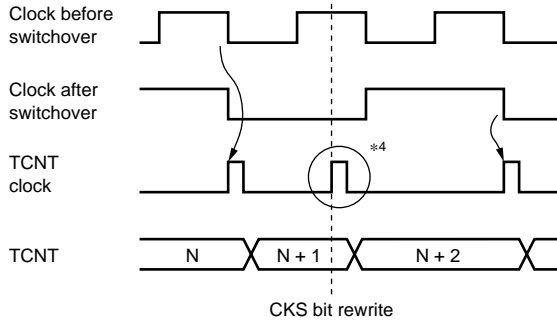


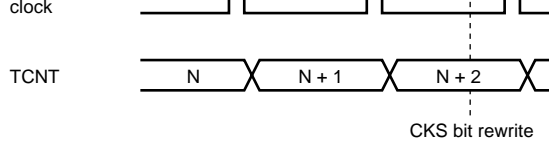


2 Clock switching from low to high level^{*2}



3 Clock switching from high to low level^{*3}





-
- Notes:
1. Includes switching from low to stop, and from stop to low.
 2. Includes switching from stop to high.
 3. Includes switching from high to stop.
 4. Generated on the assumption that the switchover is a falling edge; TCNT is incremented.

12.9.6 Mode Setting with Cascaded Connection

If the 16-bit count mode and compare-match count mode are set simultaneously, the input pulses for TCNT_0 and TCNT_1 are not generated, and thus the counters will stop operation. Simultaneous setting of these two modes should therefore be avoided.

12.9.7 Module Stop Mode Setting

TMR operation can be enabled or disabled using the module stop control register. The default setting is for TMR operation to be halted. Register access is enabled by canceling the stop mode. For details, refer to section 26, Power-Down Modes.

- Five input pins and four output pins, all of which can be designated for phase inverted output. Positive logic is assumed for all signals used within the timer connection facility.
- An edge-detection circuit is connected to the input pins, simplifying signal input detection.
- TMR_X can be used for PWM input signal decoding.
- TMR_X can be used for clamp waveform generation.
- An external clock signal divided by TMR_1 can be used as the FRT capture input.
- An internal synchronization signal can be generated using the FRT and TMR_Y.
- A signal generated/modified using an input signal and timer connection can be selected as an output.

Figure 13.1 shows a block diagram of the timer connection facility.

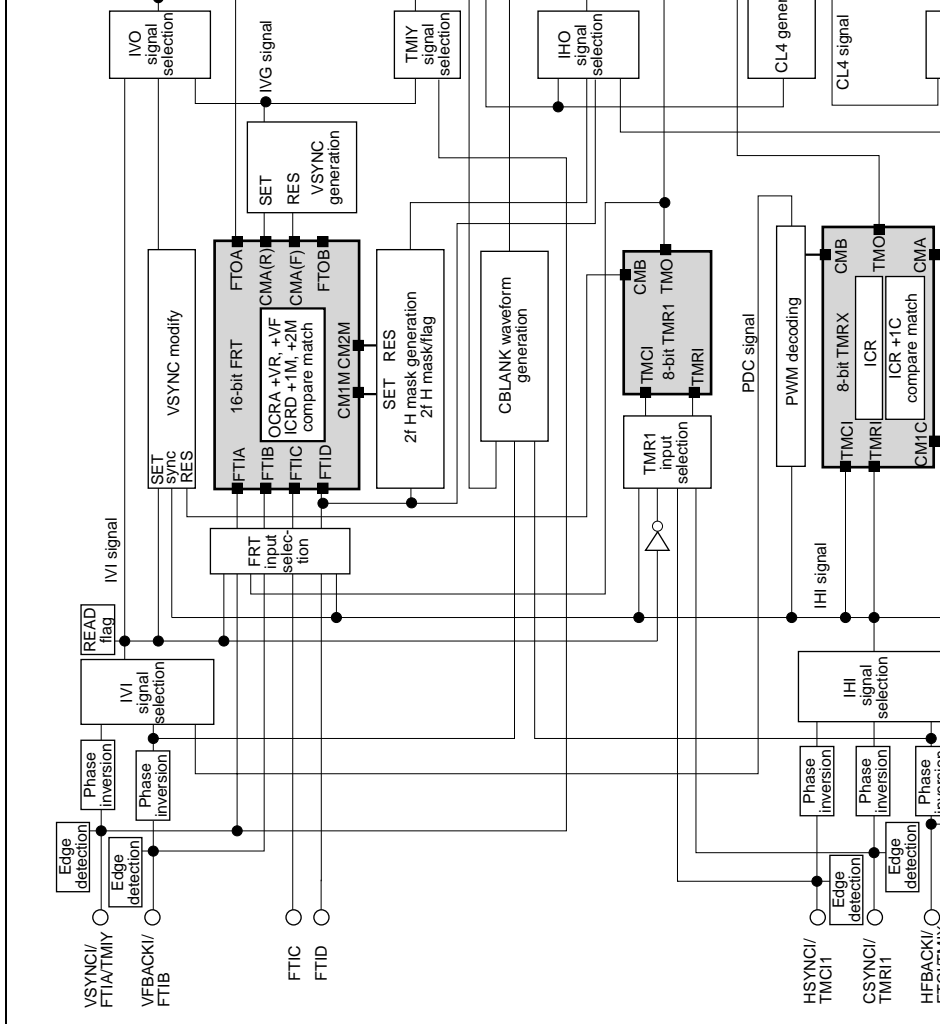


Figure 13.1 Block Diagram of Timer Connection

input pin			input pin or FTIA input pin
Horizontal synchronization signal input pin	HSYNCI	Input	Horizontal synchronization signal input pin or TMCI1 input pin
Composite synchronization signal input pin	CSYNCI	Input	Composite synchronization signal input pin or TMR11 input pin
Spare vertical synchronization signal input pin	VFBACKI	Input	Spare vertical synchronization signal input pin or FTIC input pin
Spare horizontal synchronization signal input pin	HFBACKI	Input	Spare horizontal synchronization signal input pin or FTIC input pin/TMIX input pin
Vertical synchronization signal output pin	VSYNCO	Output	Vertical synchronization signal output pin or FTOA output pin
Horizontal synchronization signal output pin	HSYNCO	Output	Horizontal synchronization signal output pin or TMO1 output pin
Clamp waveform output pin	CLAMPO	Output	Clamp waveform output pin or FTIC input pin
Blanking waveform output pin	CBLANK	Output	Blanking waveform output pin

13.3 Register Descriptions

The timer connection has the following registers.

- Timer connection register I (TCONRI)
- Timer connection register O (TCONRO)
- Timer connection register S (TCONRS)
- Edge sense register (SEDGR)

signals.

- Mode

00: No signal

01: S-on-G mode

10: Composite mode

11: Separate mode

- IHI Signal

00: HFBACKI input

01: CSYNCI input

1X: HSYNCI input

- IVI Signal

00: VFBACKI input

01: PDC input

10: PDC input

11: VSYNCI input

5	SCONE	0	R/W	Synchronization Signal Connection Enable Selects the signal source of the FRT FTI in TMR_1 TMI1 input and TMC11/TMR11 input details, see table 13.2.
---	-------	---	-----	---

1, the contents of TCNT at those points are written into TICRR and TICRF, respectively, and the contents of TICNT is cleared to 0.

[Clearing condition]

When a rising edge followed by a falling edge is detected on TMRIX

[Setting condition]

When 1 is written in ICST after reading ICST

- 0: The HFBACKI pin state is used directly as the HFBACKI input
- 1: The HFBACKI pin state is inverted before use as the HFBACKI input
- VFINV
- 0: The VFBACKI pin state is used directly as the VFBACKI input
- 1: The VFBACKI pin state is inverted before use as the VFBACKI input
- HIINV
- 0: The HSYNCI and CSYNCI pin states are used directly as the HSYNCI and CSYNCI inputs
- 1: The HSYNCI and CSYNCI pin states are inverted before use as the HSYNCI and CSYNCI inputs
- VIINV
- 0: The VSYNCI pin state is used directly as the VSYNCI input
- 1: The VSYNCI pin state is inverted before use as the VSYNCI input

Legend:

X: Don't care

Table 13.2 Synchronization Signal Connection Enable

Bit 5	Mode	Description				
		FTIA	FTIB	FTIC	FTID	TMC11
0	Normal connection (Initial value)	FTIA input	FTIB input	FTIC input	FTID input	TMC11 input
1	Synchronization signal connection mode	IVI signal	TMO1 signal	VFBACKI input	IHI signal	IHI signal

(CLAMPO), and blanking waveform (CBLANK). If output is disabled, the state of the relevant pin is determined by port DR and DDR, FRT, TMR, and PWM settings.

Output enabling/disabling control does not affect port, FRT, or TMR input functions, but some TMR input signal sources are determined by the SCONE bit in TCONRI.

HOE:

0: The P44/TMO1/HIRQ1/HSYNCO pin function as the P44/TMO1/HIRQ1 pin

1: The P44/TMO1/HIRQ1/HSYNCO pin function as the HSYNCO pin

VOE:

0: The P61/FTOA/CIN1/ $\overline{\text{KIN1}}$ /VSYNCO pin function as the P61/FTOA/CIN1/ $\overline{\text{KIN1}}$ pin

1: The P61/FTOA/CIN1/ $\overline{\text{KIN1}}$ /VSYNCO pin function as the VSYNCO pin

CLOE:

0: The P64/FTIC/CIN4/ $\overline{\text{KIN4}}$ /CLAMPO pin function as the P64/FTIC/CIN4/ $\overline{\text{KIN4}}$ pin

1: The P64/FTIC/CIN4/ $\overline{\text{KIN4}}$ /CLAMPO pin function as the CLAMPO pin

CBOE:

0: The P27/A15/PW15/CBLANK pin function as the P27/A15/PW15 pin

In mode 1:

1: The P27/A15/PW15/CBLANK pin function as the A15 pin

In modes 2 and 3:

1: The P27/A15/PW15/CBLANK pin function as the CBLANK pin

0: The IHO signal is used directly as the HSYNCO output

1: The IHO signal is inverted before use as the HSYNCO output

- VOINV:

0: The IVO signal is used directly as the VSYNCO output

1: The IVO signal is inverted before use as the VSYNCO output

- CLOINV:

0: The CLO signal (CL1, CL2, CL3, or CL4) is used directly as the CLAMPO output

1: The CLO signal (CL1, CL2, CL3, or CL4) is inverted before use as the CLAMPO output

- CBOINV:

0: The CBLANK signal is used directly as the CBLANK output

1: The CBLANK signal is inverted before use as the CBLANK output

0: The TMR_X registers are accessed at
H'(FF)FFF0 to H'(FF)FFF5
1: The TMR_Y registers are accessed at
H'(FF)FFF0 to H'(FF)FFF5

6	ISGENE	0	R/W	Internal Synchronization Signal Selects internal synchronization signals (IHI, IHO, and CL4 signals) as the signal sources for IVO, and CLO signals together with the HOMOD0, VOMOD1, VOMOD0, CLMOD0, and CLMOD0 bits.
5	HOMOD1	0	R/W	Horizontal Synchronization Output Mode
4	HOMOD0	0	R/W	These bits select the signal source and method for the IHO signal. <ul style="list-style-type: none"> • ISGENE = 0 00: The IHI signal (without 2fH modification) is selected 01: The IHI signal (with 2fH modification) is selected 1X: The CL1 signal is selected <ul style="list-style-type: none"> • ISGENE = 1 XX: The IHG signal is selected

- synchronization) is selected
- 10: The IVI signal (with fall modification, v synchronization) is selected
- 11: The IVI signal (with fall modification a synchronization) is selected
- ISGENE = 1
- XX: The IVG signal is selected

1	CLMOD1	0	R/W	Clamp Waveform Mode Select 1, 0
0	CLMOD0	0	R/W	These bits select the signal source for the (clamp waveform). <ul style="list-style-type: none"> • ISGENE = 0 00: The CL1 signal is selected 01: The CL2 signal is selected 1X: The CL3 signal is selected • ISGENE = 1 XX: The CL4 signal is selected

Legend:

X: Don't care

Table 13.3 Registers Accessible by TMR_X/TMR_Y

TMRX/Y	H'FFF0	H'FFF1	H'FFF2	H'FFF3	H'FFF4	H'FFF5	H'FFF6
0	TMR_X	TMR_X	TMR_X	TMR_X	TMR_X	TMR_X	TMR_X
	TCR_X	TCSR_X	TICRR	TICRF	TCNT_X	TCORC	TCORA_X
1	TMR_Y	TMR_Y	TMR_Y	TMR_Y	TMR_Y	TMR_Y	
	TCR_Y	TCSR_Y	TCORA_Y	TCORB_Y	TCNT_Y	TISR	

				[Clearing condition] When 0 is written in VEDG after reading [Setting condition] When a rising edge is detected on the VS
6	HEDG	0	R/(W) ^{*1}	HSYNCl Edge Detects a rising edge on the HSYNCl pin [Clearing condition] When 0 is written in HEDG after reading [Setting condition] When a rising edge is detected on the HS
5	CEDG	0	R/(W) ^{*1}	CSYNCl Edge Detects a rising edge on the CSYNCl pin [Clearing condition] When 0 is written in CEDG after reading [Setting condition] When a rising edge is detected on the CS
4	HFEDG	0	R/(W) ^{*1}	HFBACKI Edge Detects a rising edge on the HFBACKI pin [Clearing condition] When 0 is written in HFEDG after reading [Setting condition] When a rising edge is detected on the HF

2 PREQF 0 R/(W)^{1,2} Pre-Equalization Flag

Detects the occurrence of an IHI signal 2fH modification condition. The generation of a falling/rising edge in the IHI signal during a interval is expressed as the occurrence of a 2fH modification condition. For details, see section 2fH Modification of IHI Signal.

[Clearing condition]

When 0 is written in PREQF after reading 1.

[Setting condition]

When an IHI signal 2fH modification condition is detected.

1 IHI Undefined^{*2} R IHI Signal Level

Indicates the current level of the IHI signal. The source and phase inversion selection for the IHI signal depends on the contents of TCONRI. When TCONRI bit is 1, the input signal is positive, then maintain the IHI signal at positive phase by modifying TCONRI.

0: The IHI signal is low
1: The IHI signal is high

0 IVI Undefined^{*2} R IVI Signal Level

Indicates the current level of the IVI signal. The source and phase inversion selection for the IVI signal depends on the contents of TCONRI. When TCONRI bit is 1, the input signal is positive, then maintain the IVI signal at positive phase by modifying TCONRI.

0: The IVI signal is low
1: The IVI signal is high

Notes: 1. Only 0 can be written, to clear the flag.
2. The initial value is undefined since it depends on the pin state.

The timer counter (TCNT) in TMR_X is set to count the internal clock pulses and to be reset by the rising edge of the external reset signal (IHI signal). The value to be used as the threshold for deciding the pulse width is written in TCORB. The PWM decoder contains a delay latch that uses the IHI signal as data and compare-match signal B (CMB) as a clock, and the start of the IHI signal (the result of the pulse width decision) at the first compare-match signal B timing. TCNT is reset by the rise of the IHI signal is output as the PDC signal.

The pulse width setting using TICRR and TICRF of TMR_X can be used to determine the pulse width decision threshold. Examples of TCR and TCORB settings of TMR_X are shown in 13.4 and 13.5, and the PWM decoding timing chart is shown in figure 13.2.

Table 13.4 Examples of TCR Settings

Bit	Abbreviation	Contents	Description
7	CMIEB	0	Interrupts due to compare-match and overflow are disabled
6	CMIEA	0	
5	OVIE	0	
4 and 3	CCLR1 and CCLR0	11	TCNT is cleared by the rising edge of external reset signal (IHI signal)
2 to 0	CKS2 to CKS0	001	Incremented on internal clock (ϕ)

Table 13.5 Examples of TCORB (Pulse Width Threshold) Settings

	ϕ : 10 MHz	ϕ : 12 MHz	ϕ : 16 MHz	ϕ : 20 MHz
H'07	0.8 μ s	0.67 μ s	0.5 μ s	0.4 μ s
H'0F	1.6 μ s	1.33 μ s	1 μ s	0.8 μ s
H'1F	3.2 μ s	2.67 μ s	2 μ s	1.6 μ s
H'3F	6.4 μ s	5.33 μ s	4 μ s	3.2 μ s
H'7F	12.8 μ s	10.67 μ s	8 μ s	6.4 μ s

Counter reset
caused by
IHI signal

Counter clear
caused by
TCNT overflow

At the 2nd compare-match,
IHI signal is not tested

Figure 13.2 Timing Chart for PWM Decoding

13.4.2 Clamp Waveform Generation (CL1/CL2/CL3 Signal Generation)

The timer connection facility and TMR_X can be used to generate signals with different cycles and rising/falling edges (clamp waveforms) in synchronization with the input signal (IHI signal). Three clamp waveforms can be generated: the CL1 to CL3 signals. In addition, the CL3 signal can be generated using TMR_Y.

The CL1 signal rises simultaneously with the rise of the IHI signal, and when the CL1 signal is high, the CL2 signal rises simultaneously with the fall of the IHI signal. The fall of both CL1 and CL2 signals can be specified by TCORA. The rise of the CL3 signal can be specified to be simultaneous with the sampling of the fall of the IHI signal using the system clock, and the CL3 signal can be specified by TCORC. The CL3 signal can also fall when the IHI signal rises.

TCNT in TMR_X is set to count internal clock pulses and to be cleared on the rising edge of the external reset signal (IHI signal).

The value to be used as the CL1 signal pulse width is written in TCORA. Write a value of 1 or more in TCORA when internal clock ϕ is selected as the TMR_X counter clock, and a value of H'01 or more when $\phi/2$ is selected. When internal clock ϕ is selected, the CL1 signal pulse width is (TCORA set value + 3 \pm 0.5). When the CL2 signal is used, the setting must be made so that this pulse width is greater than the IHI signal pulse width.

The value to be used as the CL3 signal pulse width is written in TCORC. TICR in TMR_X captures the value of TCNT at the inverse of the external reset signal edge (in this case, the falling edge of the IHI signal). The timing of the fall of the CL3 signal is determined by the su

Both the rise and the fall of the CL3 signal are synchronized with the system clock and its pulse width is fixed, but there is a variation in the phase relationship with the IHI signal equal to the resolution of the system clock.

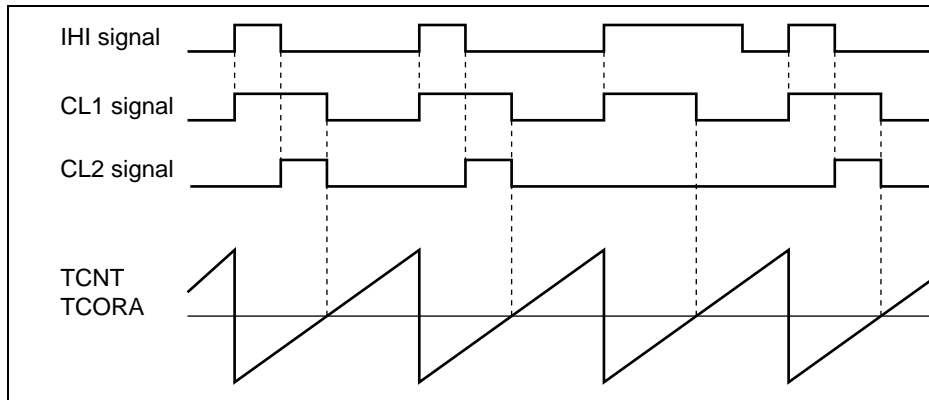


Figure 13.3 Timing Chart for Clamp Waveform Generation (CL1 and CL2)

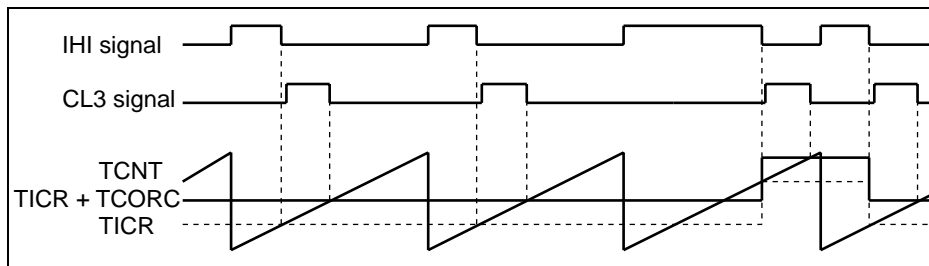


Figure 13.4 Timing Chart for Clamp Waveform Generation (CL3 Signal)

(inverse of the IVI signal). The value to be used as the division factor is written in TCR. When the TMO output method is specified by the OS bits in TCSR.

Examples of TCR and TCSR settings in TMR_1, and TCR and TCSR settings in the FTM are shown in table 13.6, and the timing chart for measurement of the IVI signal and IHI signal waveform periods is shown in figure 13.5. The period of the IHI signal divided waveform is given by $(ICRD(3) - ICRD(2)) \times \text{resolution}$.

	2 to 0	CKS2 to CKS0	101	TCNT is incremented on the falling edge of the external clock (the IVI signal)
TCSR in TMR_1	3 to 0	OS3 to OS0	0011	Not changed by compare-match B, and output inverted by compare-match A (toggle output): Division factor = 256
			1001	When TCORB < TCORA on compare-match B, and on compare-match A: Division factor = 256
TCR in FRT	6	IEDGB	0/1	0: FRC value is transferred to TCNT on falling edge of input B (IHL divided signal waveform)
				1: FRC value is transferred to TCNT on rising edge of input B (IHL divided signal waveform)
	1 and 0	CKS1 and CKS0	01	FRC is incremented on input clock: $\phi/8$
TCSR in FRT	0	CCLRA	0	FRC clearing is disabled

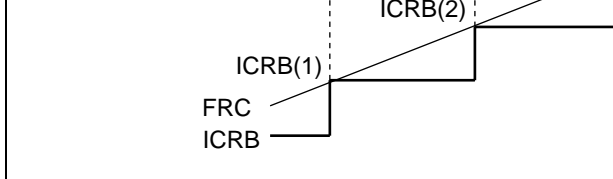


Figure 13.5 Timing Chart for Measurement of IHI Signal and IHI Signal Divided Waveform Periods

13.4.4 2fH Modification of IHI Signal

By using the timer connection facility and FRT, even if there is a part of the IHI signal the frequency, this can be eliminated. In order for this function to operate properly, the of the IHI signal must be approximately 30% or less, or approximately 70% or above.

The 8-bit OCRDM contents or twice the OCRDM contents can be added automatically captured in ICRD in the FRT, and compare-matches generated at these points. The interval between the two compare-matches is called a mask interval. A value equivalent to approximately 1/3 the IHI signal period is written in OCRDM. ICRD is set so that capture is performed on the rise of the IHI signal.

Since the IHI signal supplied to the IHO signal selection circuit is normally set on the rise of the IHI signal and reset on the fall, its waveform is the same as that of the original IHI signal. When 2fH modification is selected, IHI signal edge detection is disabled during mask intervals. IHO signal edge detection is also disabled during these intervals.

Examples of TCR, TCSR, TOCR, and OCRDM settings in the FRT are shown in table 13.5. The 2fH modification timing chart is shown in figure 13.6.

TOCR in FRT	7	ICRDMS	1	ICRD is set to the operation which OCRDM is used
OCRDM in FRT	7 to 0	OCRDM7 to OCRDM0	H'01 to H'FF	Specifies the period during which ICRD operation is masked

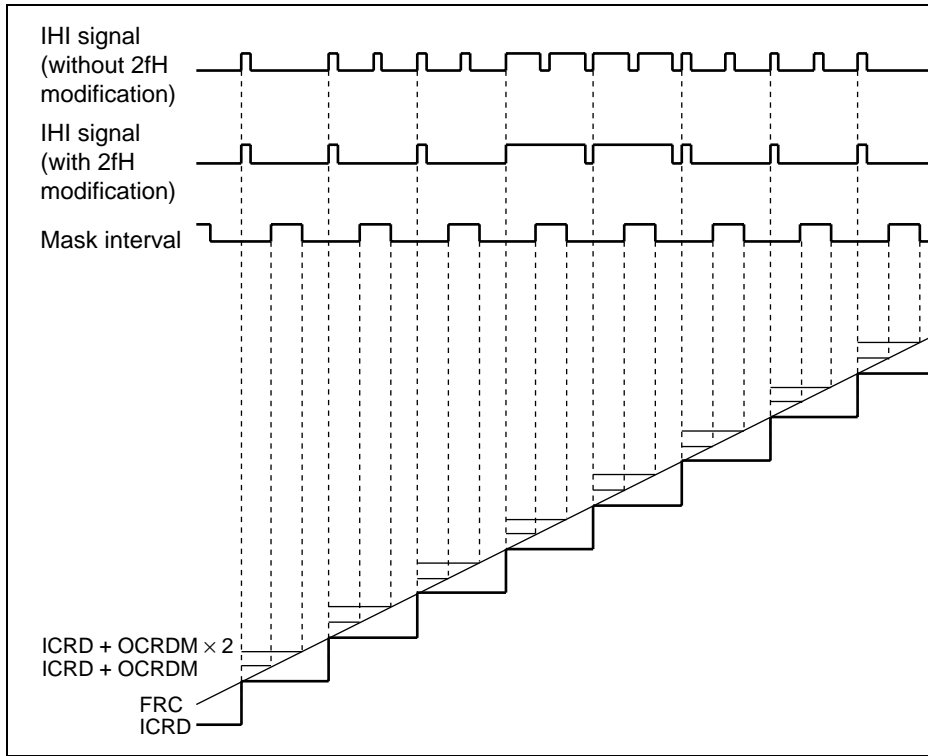


Figure 13.6 2fH Modification Timing Chart

written in TCORB.

Since the IVO signal supplied to the IVO signal selection circuit is normally set on the rise of the IVO signal and reset on the fall, its waveform is the same as that of the original IVO signal. If fall modification is selected, a reset is performed on a TMR_1 TCORB compare-match event in TMR_1.

The fall of the waveform generated in this way can be synchronized with the rise of the IVO signal regardless of whether or not fall modification is selected.

Examples of TCR, TCSR, and TCORB settings in TMR_1 are shown in table 13.8, and the IVO modification/IHI synchronization timing chart is shown in figure 13.7.

	2 to 0	CKS2 to CKS0	101	TCNT is incremented on the rise of the external clock (IHI signal)
TCSR in TMR_1	3 to 0	OS3 to OS0	0011	Not changed by compare-match output inverted by compare-match (toggle output)
			1001	When TCORB < TCORA, 1 output of compare-match B, 0 output of compare-match A
TCORB in TMR_1			H'03 (example)	Compare-match on the 4th (example) rise of the IHI signal after the inverse of the IVI signal

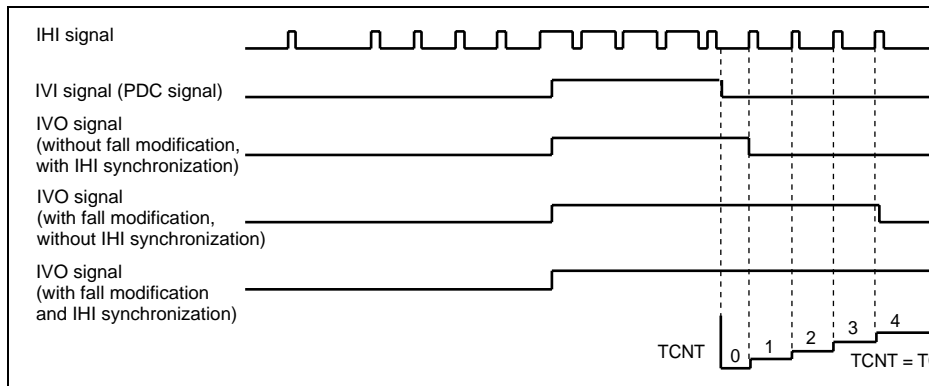


Figure 13.7 Fall Modification and IHI Synchronization Timing Chart

OCRAR or OCRAF, alternately, each time a compare-match occurs. A value corresponding to the 0 interval of the IVG signal is written in OCRAR, and a value corresponding to the 1 interval of the IVG signal is written in OCRAF. The IVG signal is set by a compare-match after an OCRAR addition, and reset by a compare-match after an OCRAF addition.

The IHG signal is the TMR_Y timer output. TMR_Y is set to count internal clock pulses. TMR_Y can be cleared on a TCORA compare-match, to fix the period and set the timer output. TCR is set so as to reset the timer output. The IVG signal is connected as the TMR_Y reset input (TMR_YRST) and the rise of the IVG signal can be treated in the same way as a TCORA compare-match.

The CL4 signal is a waveform that rises within one system clock period after the fall of the IVG signal, and has an interval of 1 for 6 system clock periods.

Examples of TCR, TCSR, TCORA, and TCORB settings in TMR_Y, and TCR, OCRA, OCRAF, and TOCR settings in the FRT are shown in table 13.9, and the IHG signal/IVG signal timing chart is shown in figure 13.8.

	2 to 0	CKS2 to CKS0	001	TCNT is incremented on inter $\phi/4$	
TCSR in TMR_Y	3 to 0	OS3 to OS0	0110	0 output on compare-match E 1 output on compare-match A	
TCORA in TMR_Y			H'3F (example)	IHG signal period = $\phi \times 256$	
TCORB in TMR_Y			H'03 (example)	IHG signal 1 interval = $\phi \times 16$	
TCR in FRT	1 and 0	CKS1 and CKS0	01	FRC is incremented on inter	
OCRAR in FRT			H'7FEF (example)	IVG signal 0 interval = $\phi \times 262016$	IVG sign $\phi \times 262$ times IH
OCRAF in FRT			H'000F (example)	IVG signal 1 interval = $\phi \times 128$	
TOCR in FRT	6	OCRAMS	1	OCRA is set to the operating which OCRAR and OCRAF a	

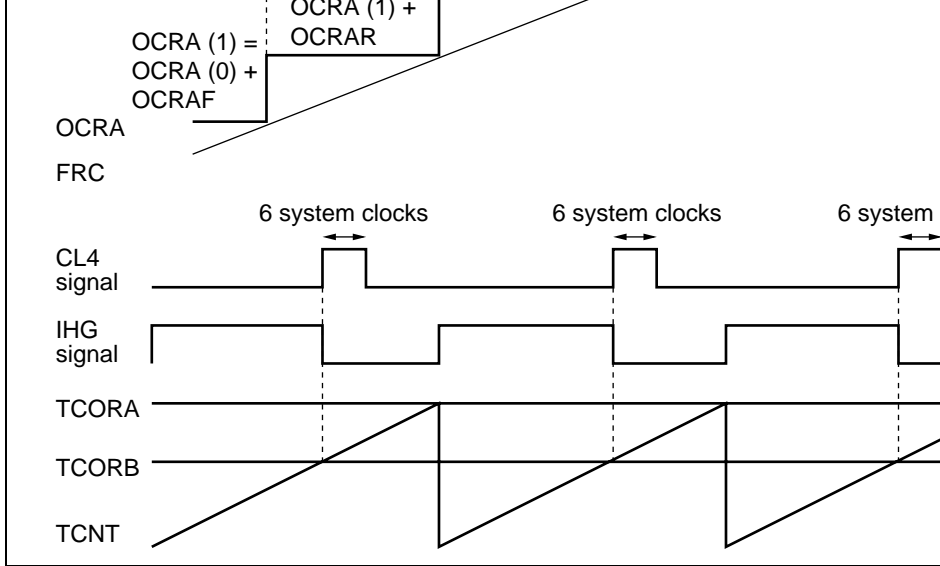


Figure 13.8 IVG Signal/IHG Signal/CL4 Signal Timing Chart

No signal	HFBACKI input	IHI signal (without 2fH modification)	HFBACKI input is output directly
		IHI signal (with 2fH modification)	Meaningless unless there is a double-f part in the HFBACKI input
		CL1 signal	HFBACKI input 1 interval is changed before output
		IHG signal	Internal synchronization signal is output directly
S-on-G mode	CSYNCl input	IHI signal (without 2fH modification)	CSYNCl input (composite synchronization signal) is output directly
		IHI signal (with 2fH modification)	Double-frequency part of CSYNCl input (composite synchronization signal) is eliminated before output
		CL1 signal	CSYNCl input (composite synchronization signal) horizontal synchronization signal part is changed before output
		IHG signal	Internal synchronization signal is output directly
Composite mode	HSYNCl input	IHI signal (without 2fH modification)	HSYNCl input (composite synchronization signal) is output directly
		IHI signal (with 2fH modification)	Double-frequency part of HSYNCl input (composite synchronization signal) is eliminated before output
		CL1 signal	HSYNCl input (composite synchronization signal) horizontal synchronization signal part is changed before output
		IHG signal	Internal synchronization signal is output directly
Separate mode	HSYNCl input	IHI signal (without 2fH modification)	HSYNCl input (horizontal synchronization signal) is output directly
		IHI signal (with 2fH modification)	Meaningless unless there is a double-f part in the HSYNCl input (horizontal synchronization signal)
		CL1 signal	HSYNCl input (horizontal synchronization signal) interval is changed before output
		IHG signal	Internal synchronization signal is output directly

No signal	VFBACKI input	IVI signal (without fall modification or IHI synchronization)	VFBACKI input is output directly
		IVI signal (without fall modification, with IHI synchronization)	Meaningless unless VFBACKI input synchronized with HFBACKI input
		IVI signal (with fall modification, without IHI synchronization)	VFBACKI input fall is modified before output
		IVI signal (with fall modification and IHI synchronization)	VFBACKI input fall is modified and synchronized with HFBACKI input output
		IVG signal	Internal synchronization signal is output
S-on-G mode or composite mode	PDC signal	IVI signal (without fall modification or IHI synchronization)	CSYNCl/HSYNCl input (composite synchronization signal) vertical synchronization signal part is separated before output
		IVI signal (without fall modification, with IHI synchronization)	CSYNCl/HSYNCl input (composite synchronization signal) vertical synchronization signal part is separated, signal is synchronized with CSYNCl input before output
		IVI signal (with fall modification, without IHI synchronization)	CSYNCl/HSYNCl input (composite synchronization signal) vertical synchronization signal part is separated, fall is modified before output
		IVI signal (with fall modification and IHI synchronization)	CSYNCl/HSYNCl input (composite synchronization signal) vertical synchronization signal part is separated, modified, and signal is synchronized with CSYNCl input before output
		IVG signal	Internal synchronization signal is output

IVI signal (with fall modification, without IHI synchronization)	fall is modified before output
IVI signal (with fall modification and IHI synchronization)	VSYNCl input (vertical synchronization fall is modified and signal is synchronized) HSYNCl input (horizontal synchronization signal) before output
IVG signal	Internal synchronization signal is

13.4.9 CBLANK Output

Using the signals generated/selected with timer connection, it is possible to generate a blanking waveform based on the composite synchronization signal (blanking waveform).

One kind of blanking waveform is generated by combining HFBACKI and VFBACKI with the phase polarity made positive by means of bits HFINV and VFINV in TCON. IVO signal.

The logic of CBLANK output waveform generation is shown in figure 13.9.

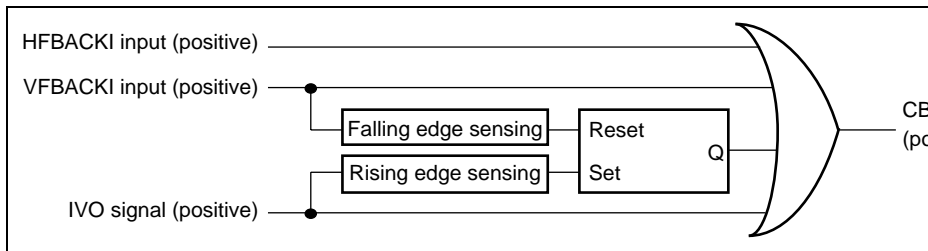


Figure 13.9 CBLANK Output Waveform Generation

timer operation, an interval timer interrupt is generated each time the counter overflows. The block diagram of the WDT_0 and WDT_1 is shown in figure 14.1.

14.1 Features

- Selectable from eight (WDT_0) or 16 (WDT_1) counter input clocks.
- Switchable between watchdog timer mode and interval timer mode

Watchdog Timer Mode:

- If the counter overflows, an internal reset or an internal NMI interrupt is generated.
- When the LSI is selected to be internally reset at counter overflow, a low level signal is output from the RESO pin if the counter overflows.

Internal Timer Mode:

- If the counter overflows, an internal timer interrupt (WOVI) is generated.

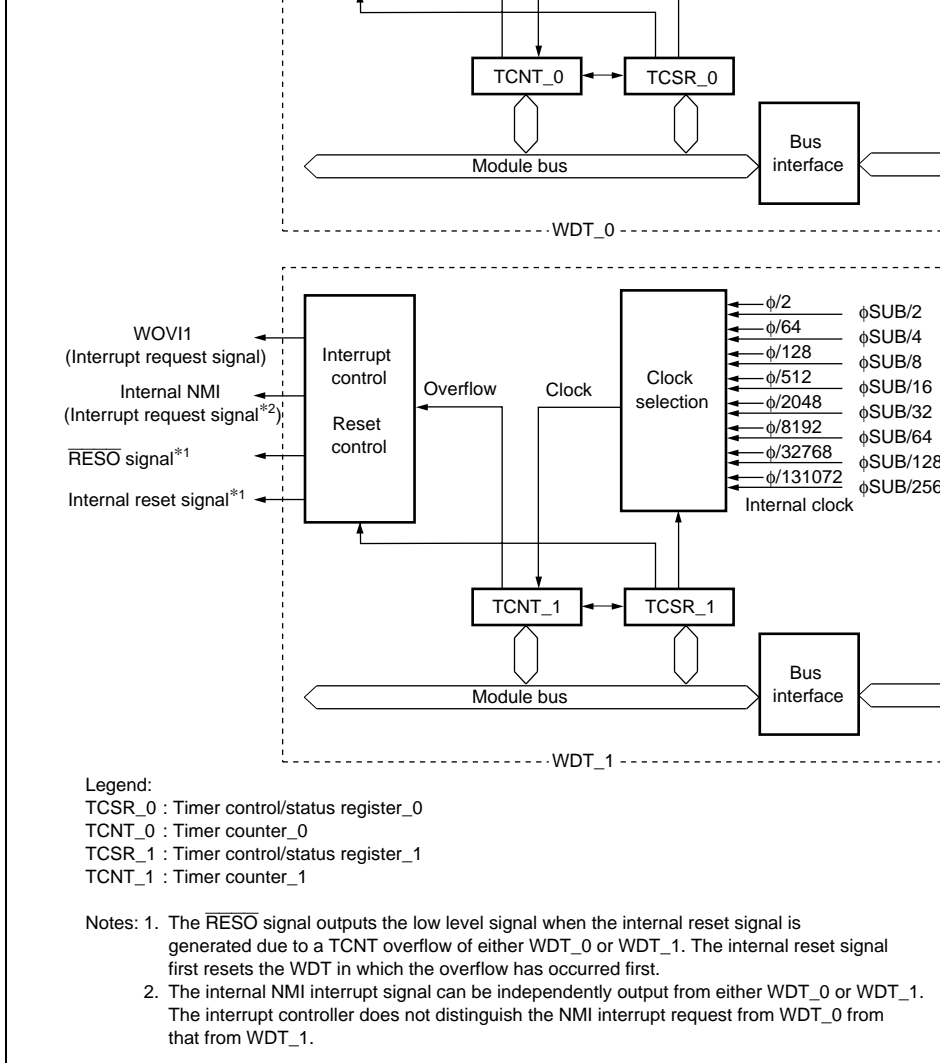


Figure 14.1 Block Diagram of WDT

14.3 Register Descriptions

The WDT has the following registers. To prevent accidental overwriting, TCSR and TCNT are to be written to in a method different from normal registers. For details, refer to section 3 Notes on Register Access. For details on the system control register, refer to section 3 Control Register (SYSCR).

- Timer counter (TCNT)
- Timer control/status register (TCSR)

14.3.1 Timer Counter (TCNT)

TCNT is an 8-bit readable/writable up-counter.

TCNT is initialized to H'00 when the TME bit in the timer control/status register (TCSR) is cleared to 0.

7	OVF	0	R/(W)	<p>Overflow Flag</p> <p>Indicates that TCNT has overflowed (changes from H'FF to H'00).</p> <p>[Setting condition]</p> <p>When TCNT overflows (changes from H'FF to H'00).</p> <p>However, when internal reset request generation is selected in watchdog timer mode, OVF is cleared automatically by the internal reset.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When TCSR is read when $OVF = 1^{*2}$, then 0 is written to OVF When 0 is written to TME
6	WT/ \overline{IT}	0	R/W	<p>Timer Mode Select</p> <p>Selects whether the WDT is used as a watchdog timer or interval timer.</p> <p>0: Interval timer mode 1: Watchdog timer mode</p>
5	TME	0	R/W	<p>Timer Enable</p> <p>When this bit is set to 1, TCNT starts counting. When this bit is cleared, TCNT stops counting and is initialized to H'00.</p>
4	—	0	R/(W)	<p>Reserved</p> <p>The initial value should not be modified.</p>
3	RST/ \overline{NMI}	0	R/W	<p>Reset or NMI</p> <p>Selects to request an internal reset or an NMI interrupt when TCNT has overflowed.</p> <p>0: An NMI interrupt is requested 1: An internal reset is requested</p>

011: $\phi/8192$ (frequency: 109.7 ms)
100: $\phi/2048$ (frequency: 52.4 ms)
101: $\phi/8192$ (frequency: 209.7 ms)
110: $\phi/32768$ (frequency: 0.84 s)
111: $\phi/131072$ (frequency: 3.36 s)

- Notes:
1. Only 0 can be written, to clear the flag.
 2. When OVF is polled with the interval timer interrupt disabled, OVF = 1 must be written at least twice.

However, when internal reset request generated, OVF is cleared automatically by the internal reset.

[Clearing conditions]

- When TCSR is read when $OVF = 1^{*2}$, then OVF is written to 0
- When 0 is written to TME

6	WT/ \overline{IT}	0	R/W	<p>Timer Mode Select</p> <p>Selects whether the WDT is used as a watchdog timer or interval timer.</p> <p>0: Interval timer mode 1: Watchdog timer mode</p>
5	TME	0	R/W	<p>Timer Enable</p> <p>When this bit is set to 1, TCNT starts counting. When this bit is cleared, TCNT stops counting and is initialized to H'00.</p>
4	PSS	0	R/W	<p>Prescaler Select</p> <p>Selects the clock source to be input to TCNT.</p> <p>0: Counts the divided cycle of ϕ-based prescaler 1: Counts the divided cycle of ϕ_{SUB}-based prescaler (PSS)</p>
3	RST/ \overline{NMI}	0	R/W	<p>Reset or NMI</p> <p>Selects to request an internal reset or an NMI interrupt when TCNT has overflowed.</p> <p>0: An NMI interrupt is requested 1: An internal reset is requested</p>

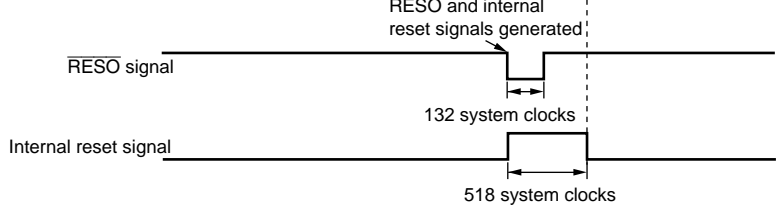
010: $\phi/128$ (frequency: 3.28 ms)
011: $\phi/512$ (frequency: 13.1 ms)
100: $\phi/2048$ (frequency: 52.4 ms)
101: $\phi/8192$ (frequency: 209.7 ms)
110: $\phi/32768$ (frequency: 0.84 s)
111: $\phi/131072$ (frequency: 3.36 s)
When PSS = 1:
000: $\phi_{SUB}/2$ (cycle: 15.6 ms)
001: $\phi_{SUB}/4$ (cycle: 31.3 ms)
010: $\phi_{SUB}/8$ (cycle: 62.5 ms)
011: $\phi_{SUB}/16$ (cycle: 125 ms)
100: $\phi_{SUB}/32$ (cycle: 250 ms)
101: $\phi_{SUB}/64$ (cycle: 500 ms)
110: $\phi_{SUB}/128$ (cycle: 1 s)
111: $\phi_{SUB}/256$ (cycle: 2 s)

-
- Notes: 1. Only 0 can be written, to clear the flag.
2. When OVF is polled with the interval timer interrupt disabled, OVF = 1 must be read at least twice.

overflows by rewriting the TCNT value (normally be writing H'00) before overflows occur. If the RST/ $\overline{\text{NMI}}$ bit of TCSR is set to 1, when the TCNT overflows, an internal reset signal is issued for 518 system clocks, and the low level signal is simultaneously output from the $\overline{\text{RESO}}$ pin for 132 states, as shown in figure 14.2. If the RST/ $\overline{\text{NMI}}$ bit is cleared to 0, when TCNT overflows, an NMI interrupt request is generated. Here, the output from the $\overline{\text{RESO}}$ pin remains high.

An internal reset request from the watchdog timer and a reset input from the $\overline{\text{RES}}$ pin are processed in the same vector. Reset source can be identified by the XRST bit status in SYSCR. If a reset caused by a signal input to the $\overline{\text{RES}}$ pin occurs at the same time as a reset caused by a watchdog timer overflow, the $\overline{\text{RES}}$ pin reset has priority and the XRST bit in SYSCR is set to 1.

An NMI interrupt request from the watchdog timer and an interrupt request from the NMI pin are processed in the same vector. Do not handle an NMI interrupt request from the watchdog timer and an interrupt request from the NMI pin at the same time.



Legend:
 WT/ $\overline{\text{IT}}$: Timer mode select bit
 TME : Timer enable bit
 OVF : Overflow flag

Note: * After the OVF bit becomes 1, it is cleared to 0 by an internal reset.
 The XRST bit is also cleared to 0.

Figure 14.2 Watchdog Timer Mode ($\overline{\text{RST}}/\overline{\text{NMI}} = 1$) Operation

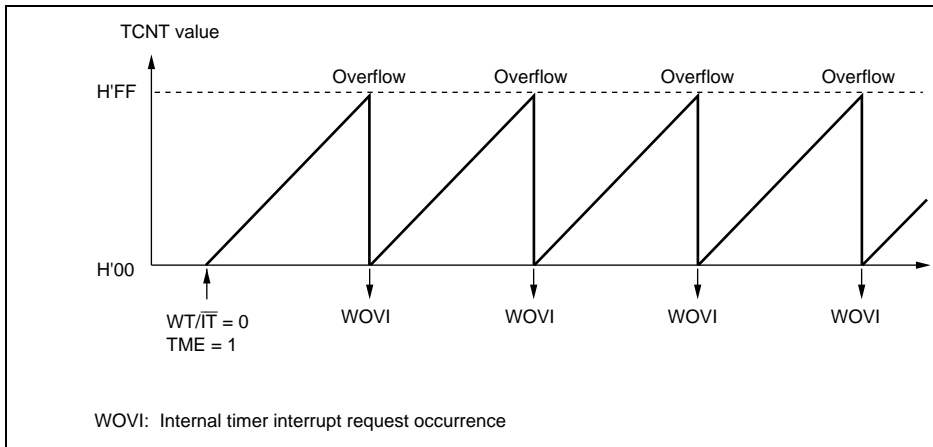


Figure 14.3 Interval Timer Mode Operation

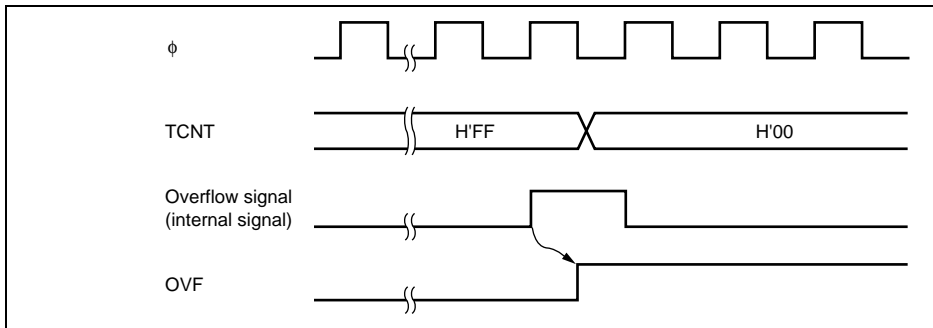


Figure 14.4 OVF Flag Set Timing

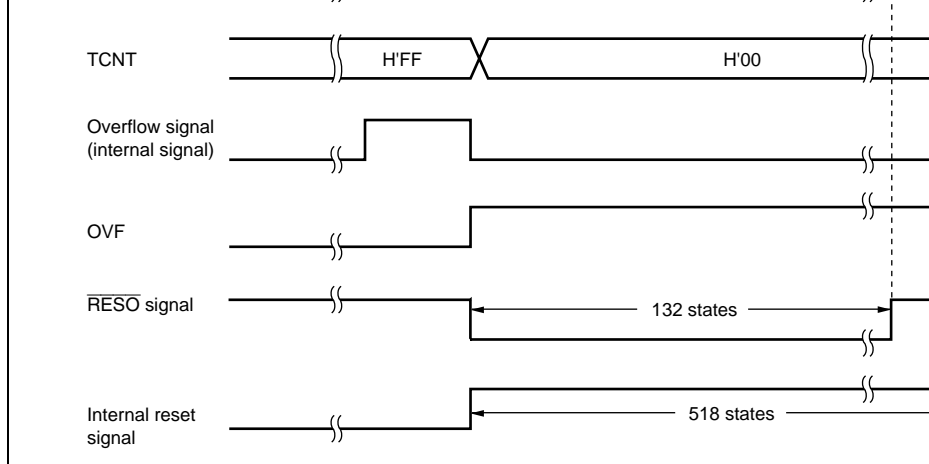


Figure 14.5 Output Timing of $\overline{\text{RESO}}$ signal

14.5 Interrupt Sources

During interval timer mode operation, an overflow generates an interval timer interrupt. The interval timer interrupt is requested whenever the OVF flag is set to 1 in TCSR. OVF is cleared to 0 in the interrupt handling routine.

When the NMI interrupt request is selected in watchdog timer mode, an NMI interrupt is generated by an overflow.

Table 14.2 WDT Interrupt Source

Name	Interrupt Source	Interrupt Flag	DTC Action
WOVI	TCNT overflow	OVF	Not possible

Writing to TCNT and TCSR (Example of WDT_0): These registers must be written by a word transfer instruction. They cannot be written to by a byte transfer instruction.

TCNT and TCSR both have the same write address. Therefore, satisfy the relative conditions shown in figure 14.6 to write to TCNT or TCSR. To write to TCNT, the upper bytes must contain the value H'5A and the lower bytes must contain the write data before the transfer instruction execution. To write to TCSR, the upper bytes must contain the value H'A5 and the lower bytes must contain the write data.

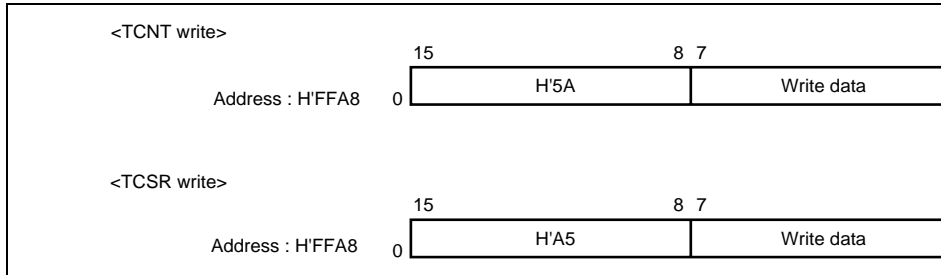


Figure 14.6 Writing to TCNT and TCSR (WDT_0)

Reading from TCNT and TCSR (Example of WDT_0): These registers are read in the same way as other registers. The read address is H'FFA8 for TCSR and H'FFA9 for TCNT.

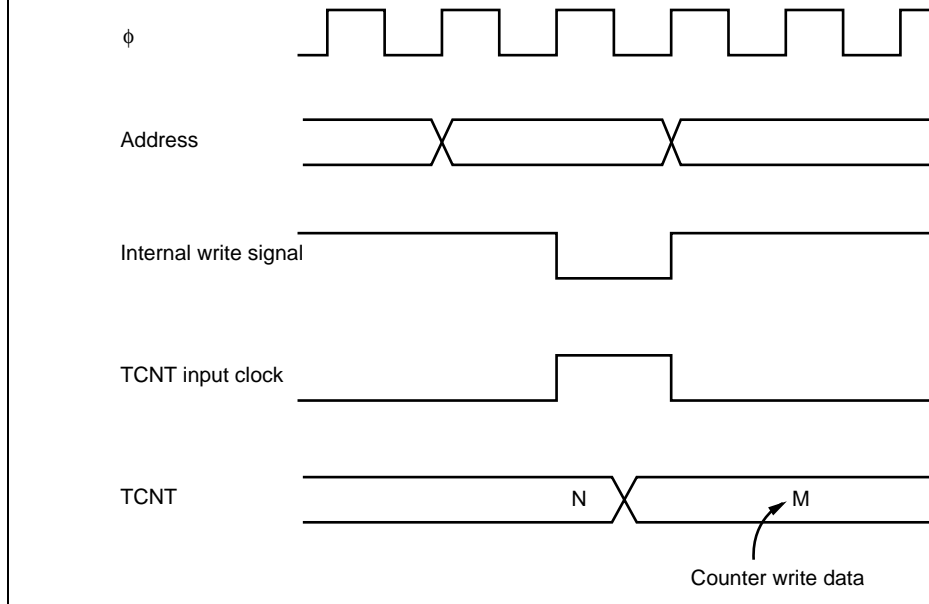


Figure 14.7 Conflict between TCNT Write and Increment

14.6.3 Changing Values of CKS2 to CKS0 Bits

If bits CKS2 to CKS0 in TCSR are written to while the WDT is operating, errors could occur during the incrementation. Software must stop the watchdog timer (by clearing the TME bit to 0) before changing the values of bits CKS2 to CKS0.

14.6.4 Switching between Watchdog Timer Mode and Interval Timer Mode

If the mode is switched from watchdog timer to interval timer, while the WDT is operating, errors could occur in the incrementation. Software must stop the watchdog timer (by clearing the TME bit to 0) before switching the mode.

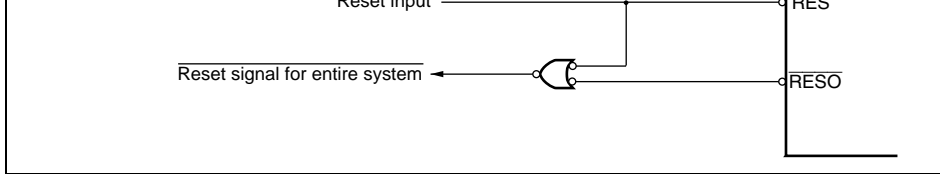


Figure 14.8 Sample Circuit for Resetting System by $\overline{\text{RESO}}$ Signal

14.6.6 Counter Values during Transitions between High-Speed, Sub-Active, and Watch Modes

When WDT_1 is used as a clock counter and is allowed to transit between high-speed mode, sub-active or watch mode, the counter does not display the correct value due to internal switching.

Specifically, when transiting from high-speed mode to sub-active or watch mode, that is, when the control clock for WDT_1 switches from the main clock to the sub-clock, the counter incrementing timing is delayed for approximately two to three clock cycles.

Similarly, when transiting from sub-active or watch mode to high-speed mode, the clock is not supplied until stabilized internal oscillation is available because the main clock oscillation is not available in sub-clock mode. The counter is therefore prevented from incrementing for the time specified by the STS2 to STS0 bits in SBYCR after internal oscillation starts, thus producing counter value differences for this time.

Special care must be taken when using WDT_1 as a clock counter. Note that no counter value difference is produced while operated in the same mode.

SCI_2 can handle communication using the waveform based on the Infrared Data Association (IrDA) standard version 1.0.

15.1 Features

- Choice of asynchronous or clocked synchronous serial communication mode
- Full-duplex communication capability
The transmitter and receiver are mutually independent, enabling transmission and reception to be executed simultaneously. Double-buffering is used in both the transmitter and receiver, enabling continuous transmission and continuous reception of serial data.
- The on-chip baud rate generator allows any bit rate to be selected
An external clock can be selected as a transfer clock source.
- Choice of LSB-first or MSB-first transfer (except in the case of asynchronous mode)
- Four interrupt sources
Four interrupt sources — transmit-end, transmit-data-empty, receive-data-full, and receive-error — that can issue requests.
The transmit-data-empty and receive-data-full interrupt sources can activate the DMA controller.

Asynchronous Mode:

- Data length: 7 or 8 bits
- Stop bit length: 1 or 2 bits
- Parity: Even, odd, or none
- Receive error detection: Parity, overrun, and framing errors
- Break detection: Break can be detected by reading the RxD pin level directly in case of a framing error

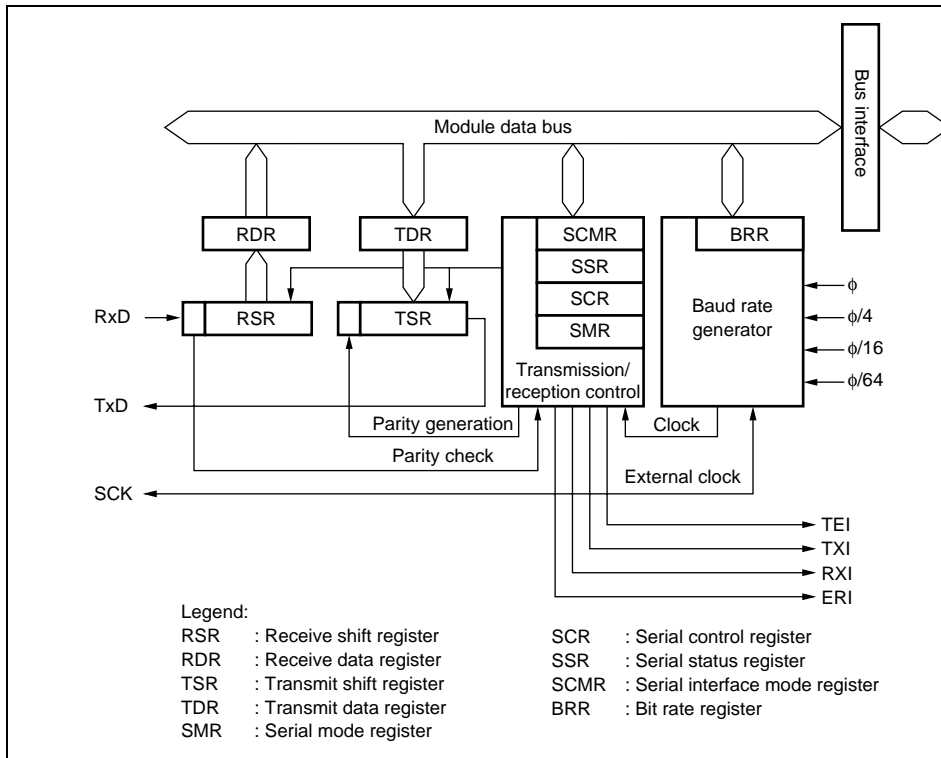


Figure 15.1 Block Diagram of SCI

	RxD0	Input	Channel 0 receive data input
	TxD0	Output	Channel 0 transmit data output
1	SCK1	Input/Output	Channel 1 clock input/output
	RxD1	Input	Channel 1 receive data input
	TxD1	Output	Channel 1 transmit data output
2	SCK2	Input/Output	Channel 2 clock input/output
	RxD2/IrRxD	Input	Channel 2 receive data input (normal/IrD)
	TxD2/IrTxD	Output	Channel 2 transmit data output (normal/IrD)

Note: * Pin names SCK, RxD, and TxD are used in the text for all channels, omitting channel designation.

15.3 Register Descriptions

The SCI has the following registers for each channel.

- Receive shift register (RSR)
- Receive data register (RDR)
- Transmit data register (TDR)
- Transmit shift register (TSR)
- Serial mode register (SMR)
- Serial control register (SCR)
- Serial status register (SSR)
- Serial interface mode register (SCMR)
- Bit rate register (BRR)
- Keyboard comparator control register (KBCOMP)

RDR is an 8-bit register that stores receive data. When the SCI has received one frame of data, it transfers the received serial data from RSR to RDR where it is stored. After this, the SCI can receive the next data. Since RSR and RDR function as a double buffer in this way, continuous receive operations can be performed. After confirming that the RDRF bit in SSR is set to 1, the CPU can read RDR for only once. RDR cannot be written to by the CPU. RDR is initialized to H'00.

15.3.3 Transmit Data Register (TDR)

TDR is an 8-bit register that stores transmit data. When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission. The double-buffered structures of TDR and TSR enable continuous serial transmission. If the next transmit data has already been written to TDR when one frame of data is transmitted, the SCI transfers the next transmit data to TSR to continue transmission. Although TDR can be read from or written to by the CPU at all times, to achieve reliable serial transmission, write transmit data to TDR for only once, confirming that the TDRE bit in SSR is set to 1. TDR is initialized to H'FF.

15.3.4 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data. To perform serial data transmission, the SCI transfers transmit data from TDR to TSR, then sends the data to the TxD pin. TSR cannot be directly accessed by the CPU.

6	CHR	0	R/W	<p>Character Length (enabled only in asynchronous mode)</p> <p>0: Selects 8 bits as the data length.</p> <p>1: Selects 7 bits as the data length. LSB-first and the MSB of TDR is not transmitted in transmission.</p> <p>In clocked synchronous mode, a fixed data length of 8 bits is used.</p>
5	PE	0	R/W	<p>Parity Enable (enabled only in asynchronous mode)</p> <p>When this bit is set to 1, the parity bit is added to the data to transmit data before transmission, and the parity bit is checked in reception. For a multiprocessor system, parity bit addition and checking are not performed regardless of the PE bit setting.</p>
4	O \bar{E}	0	R/W	<p>Parity Mode (enabled only when the PE bit is set to 1 in asynchronous mode)</p> <p>0: Selects even parity.</p> <p>1: Selects odd parity.</p>
3	STOP	0	R/W	<p>Stop Bit Length (enabled only in asynchronous mode)</p> <p>Selects the stop bit length in transmission.</p> <p>0: 1 stop bit</p> <p>1: 2 stop bits</p> <p>In reception, only the first stop bit is checked. If the second stop bit is 0, it is treated as the start of the next transmit frame.</p>

rate generator.

00: ϕ clock (n = 0)

01: $\phi/4$ clock (n = 1)

10: $\phi/16$ clock (n = 2)

11: $\phi/64$ clock (n = 3)

For the relation between the bit rate register and the baud rate, see section 15.3.9, Bit Rate Register (BRR). n is the decimal display of n in BRR.

15.3.6 Serial Control Register (SCR)

SCR is a register that performs enabling or disabling of SCI transfer operations and interrupt requests, and selection of the transfer clock source. For details on interrupt requests, refer to section 15.8, Interrupt Sources.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When this bit is set to 1, a TXI interrupt request is enabled.
6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled.
5	TE	0	R/W	Transmit Enable When this bit is set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.

2	TEIE	0	R/W	Transmit End Interrupt Enable When this bit is set to 1, a TEI interrupt re-enabled.
1	CKE1	0	R/W	Clock Enable 1, 0
0	CKE0	0	R/W	These bits select the clock source and SC function. Asynchronous mode 00: Internal clock (SCK pin functions as I/O port.) 01: Internal clock (Outputs a clock of the same frequency as from the SCK pin.) 1X: External clock (Inputs a clock with a frequency 16 times t from the SCK pin.) Clocked synchronous mode 0X: Internal clock (SCK pin functions as cl 1X: External clock (SCK pin functions as c

Legend:

X: Don't care

[Setting conditions]

- When the TE bit in SCR is 0
- When data is transferred from TDR to TDR is ready for data write

[Clearing conditions]

- When 0 is written to TDRE after reading the DTC to read data from TDR
TDRE = 1
- When a TXI interrupt request is issued to the DTC to write data to TDR

6	RDRF	0	R/(W)*	Receive Data Register Full Indicates that receive data is stored in RDR. [Setting condition] When serial reception ends normally and data is transferred from RSR to RDR [Clearing conditions] <ul style="list-style-type: none">• When 0 is written to RDRF after reading the DTC to read data from RDR RDRF = 1• When an RXI interrupt request is issued to the DTC to read data from RDR The RDRF flag is not affected and retains its previous value when the RE bit in SCR is 0.
5	ORER	0	R/(W)*	Overrun Error [Setting condition] When the next data is received while RDR is full. [Clearing condition] When 0 is written to ORER after reading the DTC to read data from RDR. ORER = 1

3	PER	0	R/(W)*	Parity Error [Setting condition] When a parity error is detected during re [Clearing condition] When 0 is written to PER after reading f
2	TEND	1	R	Transmit End [Setting conditions] <ul style="list-style-type: none"> • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of t a 1-byte serial transmit character [Clearing conditions] <ul style="list-style-type: none"> • When 0 is written to TDRE after read TDRE = 1 • When a TXI interrupt request is issu the DTC to write data to TDR
1	MPB	0	R	Multiprocessor Bit MPB stores the multiprocessor bit in the frame. When the RE bit in SCR is clear previous state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer MPBT stores the multiprocessor bit to b the transmit frame.

Note: * Only 0 can be written, to clear the flag.

3	SDIR	0	R/W	<p>Data Transfer Direction</p> <p>0: TDR contents are transmitted with LSB first in RDR. Receive data is stored as LSB first in RDR.</p> <p>1: TDR contents are transmitted with MSB first in RDR. Receive data is stored as MSB first in RDR.</p> <p>The SDIR bit is valid only when the 8-bit data format is used for transmission/reception; when the 16-bit data format is used, data is always transmitted/received with LSB-first.</p>
2	SINV	0	R/W	<p>Data Invert</p> <p>Specifies inversion of the data logic level. The parity bit does not affect the logic level of the parity bit. When the parity bit is inverted, invert the SMR.</p> <p>0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR.</p> <p>1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR.</p>
1	—	1	R	<p>Reserved</p> <p>This bit is always read as 1 and cannot be written.</p>
0	SMIF	0	R/W	<p>Serial Communication Interface Mode Selection</p> <p>0: Normal asynchronous or clocked synchronous mode</p> <p>1: Reserved mode</p>

Mode	Bit Rate	Error
Asynchronous mode	$B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N+1)}$	$\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} \right\}$
Clocked synchronous mode	$B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N+1)}$	—

Legend:

B: Bit rate (bit/s)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

ϕ : Operating frequency (MHz)

n: Determined by the SMR settings shown in the following table.

SMR Setting		
CKS1	CKS0	n
0	0	0
0	1	1
1	0	2
1	1	3

Table 15.3 shows sample N settings in BRR in normal asynchronous mode. Table 15.4 shows the maximum bit rate settable for each frequency. Table 15.6 shows sample N settings in BRR in clocked synchronous mode. Tables 15.5 and 15.7 show the maximum bit rates with external clock input.

300	0	207	0.16	0	217	0.21	0	255	0.00	1	1
600	0	103	0.16	0	108	0.21	0	127	0.00	0	1
1200	0	51	0.16	0	54	-0.70	0	63	0.00	0	7
2400	0	25	0.16	0	26	1.14	0	31	0.00	0	3
4800	0	12	0.16	0	13	-2.48	0	15	0.00	0	1
9600	—	—	—	0	6	-2.48	0	7	0.00	0	9
19200	—	—	—	—	—	—	0	3	0.00	0	4
31250	0	1	0.00	—	—	—	—	—	—	0	2
38400	—	—	—	—	—	—	0	1	0.00	—	—

Bit Rate (bit/s)	Operating Frequency ϕ (MHz)										
	3.6864			4			4.9152				
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N
110	2	64	0.70	2	70	0.03	2	86	0.31	2	8
150	1	191	0.00	1	207	0.16	1	255	0.00	2	6
300	1	95	0.00	1	103	0.16	1	127	0.00	1	1
600	0	191	0.00	0	207	0.16	0	255	0.00	1	6
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	1
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	6
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	3
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	1
19200	0	5	0.00	—	—	—	0	7	0.00	0	7
31250	—	—	—	0	3	0.00	0	4	-1.70	0	4
38400	0	2	0.00	—	—	—	0	3	0.00	0	3

Legend:

—: Can be set, but there will be a degree of error.

Note: Make the settings so that the error does not exceed 1%.

600	1	77	0.16	1	79	0.00	1	95	0.00	1
1200	0	155	0.16	0	159	0.00	0	191	0.00	0
2400	0	77	0.16	0	79	0.00	0	95	0.00	0
4800	0	38	0.16	0	39	0.00	0	47	0.00	0
9600	0	19	-2.34	0	19	0.00	0	23	0.00	0
19200	0	9	-2.34	0	9	0.00	0	11	0.00	0
31250	0	5	0.00	0	5	2.40	—	—	—	0
38400	0	4	-2.34	0	4	0.00	0	5	0.00	—

Bit Rate (bit/s)	Operating Frequency ϕ (MHz)									
	9.8304			10			12			n
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	
110	2	174	-0.26	2	177	-0.25	2	212	0.03	2
150	2	127	0.00	2	129	0.16	2	155	0.16	2
300	1	255	0.00	2	64	0.16	2	77	0.16	2
600	1	127	0.00	1	129	0.16	1	155	0.16	1
1200	0	255	0.00	1	64	0.16	1	77	0.16	1
2400	0	127	0.00	0	129	0.16	0	155	0.16	0
4800	0	63	0.00	0	64	0.16	0	77	0.16	0
9600	0	31	0.00	0	32	-1.36	0	38	0.16	0
19200	0	15	0.00	0	15	1.73	0	19	-2.34	0
31250	0	9	-1.70	0	9	0.00	0	11	0.00	0
38400	0	7	0.00	0	7	1.73	0	9	-2.34	0

Legend:

—: Can be set, but there will be a degree of error.

Note: Make the settings so that the error does not exceed 1%.

600	1	181	0.16	1	191	0.00	1	207	0.16	1	217
1200	1	90	0.16	1	95	0.00	1	103	0.16	1	110
2400	0	181	0.16	0	191	0.00	0	207	0.16	0	217
4800	0	90	0.16	0	95	0.00	0	103	0.16	0	110
9600	0	45	-0.93	0	47	0.00	0	51	0.16	0	55
19200	0	22	-0.93	0	23	0.00	0	25	0.16	0	27
31250	0	13	0.00	0	14	-1.70	0	15	0.00	0	16
38400	—	—	—	0	11	0.00	0	12	0.16	0	13

Bit Rate (bit/s)	Operating Frequency ϕ (MHz)								
	18			19.6608			20		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	79	-0.12	3	86	0.31	3	88	-0.25
150	2	233	0.16	2	255	0.00	3	64	0.16
300	2	116	0.16	2	127	0.00	2	129	0.16
600	1	233	0.16	1	255	0.00	2	64	0.16
1200	1	166	0.16	1	127	0.00	1	129	0.16
2400	0	233	0.16	0	255	0.00	1	64	0.16
4800	0	166	0.16	0	127	0.00	0	129	0.16
9600	0	58	-0.69	0	63	0.00	0	64	0.16
19200	0	28	1.02	0	31	0.00	0	32	-1.36
31250	0	17	0.00	0	19	-1.70	0	19	0.00
38400	0	14	-2.34	0	15	0.00	0	15	1.73

Legend:

—: Can be set, but there will be a degree of error.

Note: Make the settings so that the error does not exceed 1%.

3.6864	115200	0	0
4	125000	0	0
4.9152	153600	0	0
5	156250	0	0
6	187500	0	0
6.144	192000	0	0
7.3728	230400	0	0
8	250000	0	0

14	437500	0
14.7456	460800	0
16	500000	0
17.2032	537600	0
18	562500	0
19.6608	614400	0
20	625000	0

Table 15.5 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)	ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
2	0.5000	31250	9.8304	2.4576	153600
2.097152	0.5243	32768	10	2.5000	156250
2.4576	0.6144	38400	12	3.0000	187500
3	0.7500	46875	12.288	3.0720	192000
3.6864	0.9216	57600	14	3.5000	218750
4	1.0000	62500	14.7456	3.6864	230400
4.9152	1.2288	76800	16	4.0000	250000
5	1.2500	78125	17.2032	4.3008	268750
6	15.000	93750	18	4.5000	281250
6.144	1.5360	96000	19.6608	4.9152	307200
7.3728	1.8432	115200	20	5.0000	312500
8	2.0000	125000			



1k	1	124	1	249	2	124	—	—	2	249	—
2.5k	0	199	1	99	1	199	1	249	2	99	2
5k	0	99	0	199	1	99	1	124	1	199	1
10k	0	49	0	99	0	199	0	249	1	99	1
25k	0	19	0	39	0	79	0	99	0	159	0
50k	0	9	0	19	0	39	0	49	0	79	0
100k	0	4	0	9	0	19	0	24	0	39	0
250k	0	1	0	3	0	7	0	9	0	15	0
500k	0	0*	0	1*	0	3	0	4	0	7	0
1M			0	0	0	1			0	3	0
2.5M							0	0*			0
5M											0

Legend:

Blank: Cannot be set.

—: Can be set, but there will be a degree of error.

*: Continuous transfer or reception is not possible.

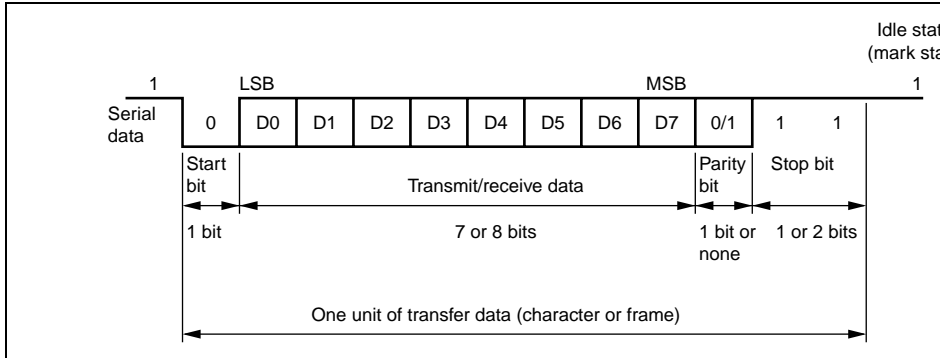
Table 15.7 Maximum Bit Rate with External Clock Input (Clock Synchronous)

ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)	ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
2	0.3333	333333.3	12	2.0000	2000000.0
4	0.6667	666666.7	14	2.3333	2333333.3
6	1.0000	1000000.0	16	2.6667	2666666.7
8	1.3333	1333333.3	18	3.0000	3000000.0
10	1.6667	1666666.7	20	3.3333	3333333.3

and RxD2 pins, respectively
 1: TxD2/IrTxD and RxD2/IrRxD pins function
 and IrRxD pins, respectively

6	IrCKS2	0	R/W	IrDA Clock Select 2 to 0	
5	IrCKS1	0	R/W	These bits specify the high-level width of the pulse during IrTxD output pulse encoding when IrDA function is enabled. 000: $B \times 3/16$ (B: Bit rate) 001: $\phi/2$ 010: $\phi/4$ 011: $\phi/8$ 100: $\phi/16$ 101: $\phi/32$ 110: $\phi/64$ 111: $\phi/128$	
4	IrCKS0	0	R/W		
3	KBADE	0	R/W		Bits related to the A/D converter
2	KBCH2	0	R/W		For details, refer to section 21.3.4, Keyboard Comparator Control Register (KBCOMP).
1	KBCH1	0	R/W		
0	KBCH0	0	R/W		

receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transfer and reception.



**Figure 15.2 Data Format in Asynchronous Communication
(Example with 8-Bit Data, Parity, Two Stop Bits)**

Serial Settings				Serial Transmitter Receive Format and Frame Length													
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12		
0	0	0	0	S	8-bit data								STOP				
0	0	0	1	S	8-bit data								STOP	STOP			
0	1	0	0	S	8-bit data								P	STOP			
0	1	0	1	S	8-bit data								P	STOP	STOP		
1	0	0	0	S	7-bit data							STOP					
1	0	0	1	S	7-bit data							STOP	STOP				
1	1	0	0	S	7-bit data							P	STOP				
1	1	0	1	S	7-bit data							P	STOP	STOP			
0	—	1	0	S	8-bit data								MPB	STOP			
0	—	1	1	S	8-bit data								MPB	STOP	STOP		
1	—	1	0	S	7-bit data							MPB	STOP				
1	—	1	1	S	7-bit data							MPB	STOP	STOP			

Legend:

S: Start bit

STOP: Stop bit

P: Parity bit

MPB: Multiprocessor bit

$$M = \left\{ \left(0.5 - \frac{1}{2N} \right) - \frac{D - 0.5}{N} (1 + F) - (L - 0.5) F \right\} \times 100 \quad [\%] \quad \dots \quad \text{Formula (1)}$$

- M: Reception margin (%)
- N : Ratio of bit rate to clock (N = 16)
- D : Clock duty (D = 0.5 to 1.0)
- L : Frame length (L = 9 to 12)
- F : Absolute value of clock rate deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100 \quad [\%] = 46.875 \%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

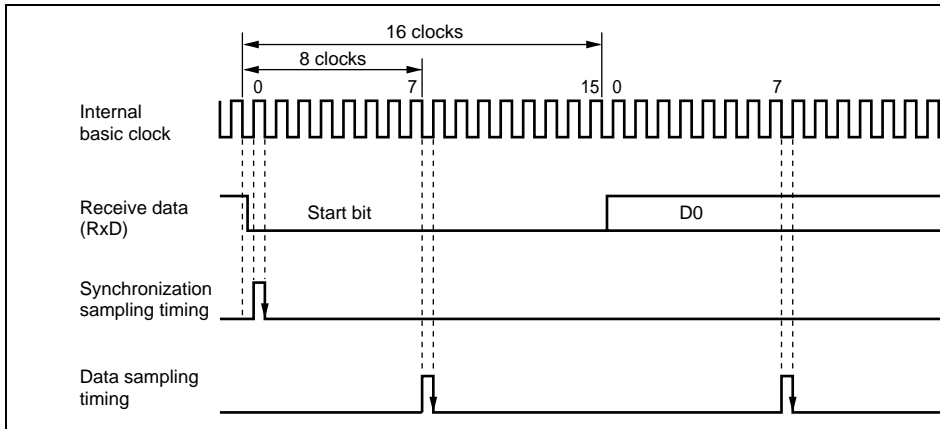


Figure 15.3 Receive Data Sampling Timing in Asynchronous Mode

rising edge of the clock is in the middle of the transmit data, as shown in figure 15.4.

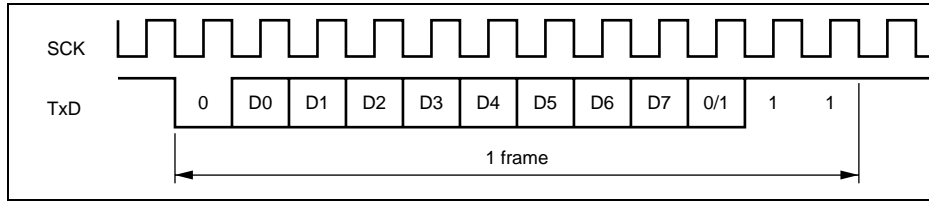


Figure 15.4 Relation between Output Clock and Transmit Data Phase (Asynchronous Mode)

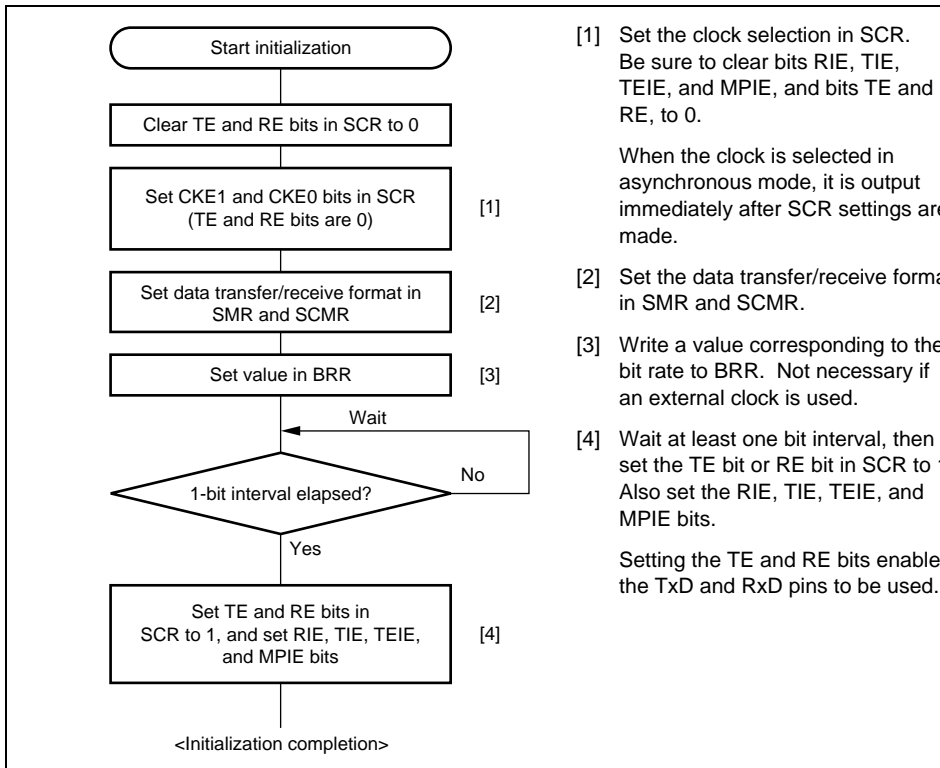


Figure 15.5 Sample SCI Initialization Flowchart

- transmission. If the TEIE bit in SCR is set to 1 at this time, a transmit data empty interrupt request (TXI) is generated. Because the TXI interrupt routine writes the next transmit data to the TDR before transmission of the current transmit data has finished, continuous transmission can be enabled.
3. Data is sent from the TxD pin in the following order: start bit, transmit data, parity, multiprocessor bit (may be omitted depending on the format), and stop bit.
 4. The SCI checks the TDRE flag at the timing for sending the stop bit.
 5. If the TDRE flag is 0, the data is transferred from TDR to TSR, the stop bit is sent, and the serial transmission of the next frame is started.
 6. If the TDRE flag is 1, the TEND flag in SSR is set to 1, the stop bit is sent, and the "transmission state" is entered in which 1 is output. If the TEIE bit in SCR is set to 1 at this time, an interrupt request is generated.

Figure 15.7 shows a sample flowchart for transmission in asynchronous mode.

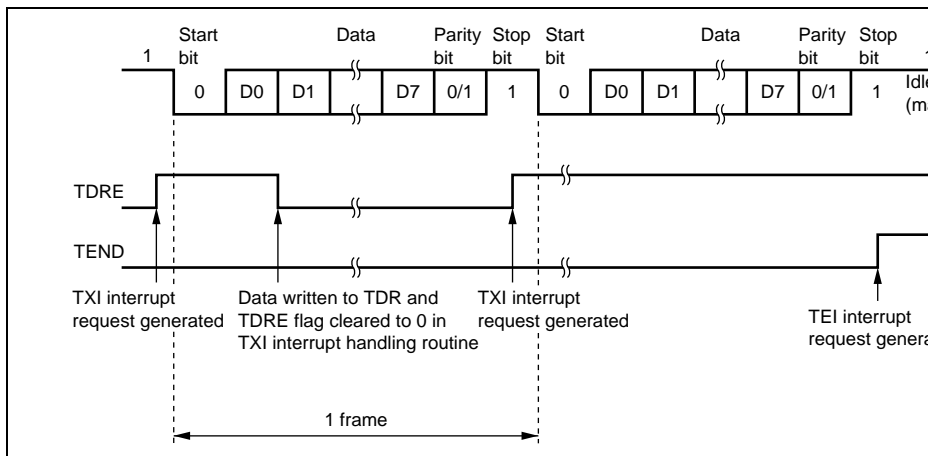
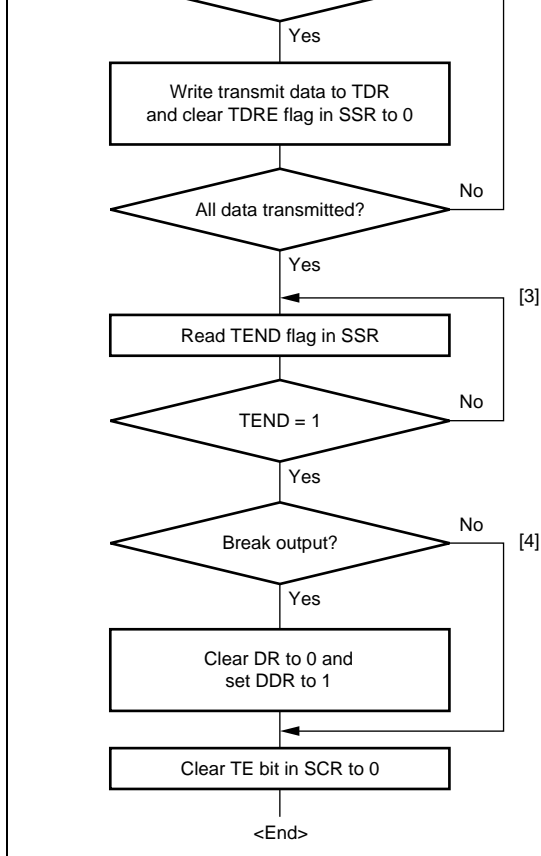


Figure 15.6 Example of SCI Transmit Operation in Asynchronous Mode (Example of 8-Bit Data, Parity, One Stop Bit)



Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:

To continue serial transmission, read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and clear the TDRE flag to 0. However, the TDRE flag is checked and cleared automatically when the DTC is initiated by a transmit data empty interrupt (TXI) request and writes data to TDR.

[4] Break output at the end of serial transmission:

To output a break in serial transmission, set DDR for the port corresponding to the TxD pin to 1, clear DR to 0, then clear the TE bit in SCR to 0.

Figure 15.7 Sample Serial Transmission Flowchart

- is still set to 1) occurs, the OREX bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The ERI interrupt request flag remains to be set to 1.
3. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
 4. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
 5. If reception finishes successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Because the RXI interrupt routine reads the receive data transferred to RDR, reception of the next receive data has finished, continuous reception can be enabled.

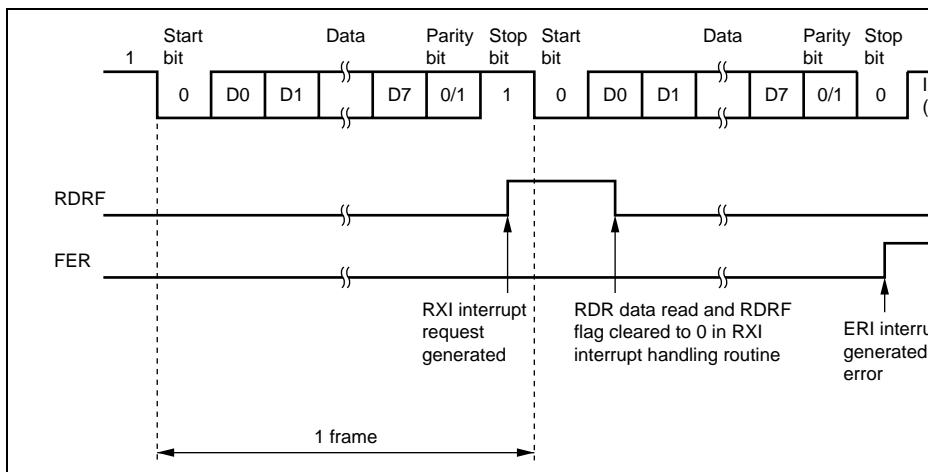
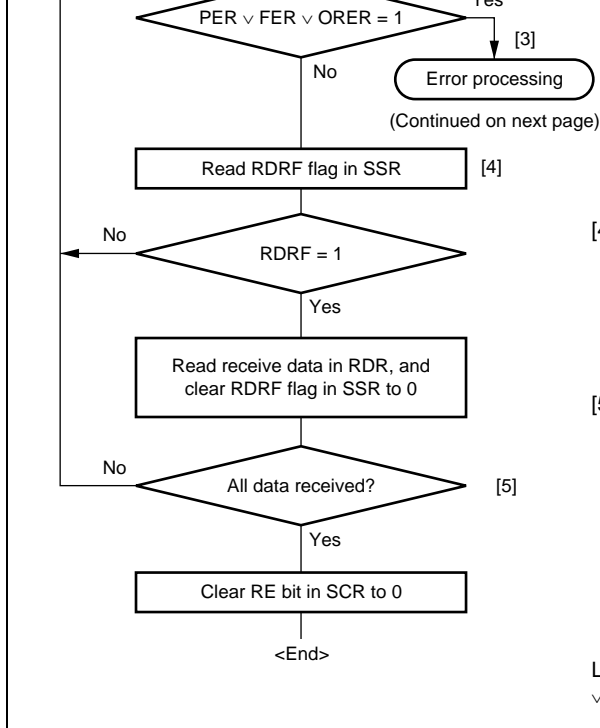


Figure 15.8 Example of SCI Receive Operation in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

RDRF*	ORER	FER	PER	Receive Data	Receive Error Type
1	1	0	0	Lost	Overrun error
0	0	1	0	Transferred to RDR	Framing error
0	0	0	1	Transferred to RDR	Parity error
1	1	1	0	Lost	Overrun error + framing error
1	1	0	1	Lost	Overrun error + parity error
0	0	1	1	Transferred to RDR	Framing error + parity error
1	1	1	1	Lost	Overrun error + framing error + parity error

Note: * The RDRF flag retains the state it had before data reception.



appropriate error processing, ensure that the ORER, PER, and FER flags are all cleared to 0. Reception can be resumed if any of these flags are cleared to 0. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the Rx pin.

[4] SCI status check and receive data read: Read SSR and check that RDRF = 1. If RDRF = 1, then read the receive data in RDR, clear the RDRF flag to 0. Transferring the RDRF flag from 0 to 1 can be identified by an RXI interrupt.

[5] Serial reception continuation procedure: To continue serial reception, before the stop bit for the current frame is received, read the RDRF flag, read RDR, and clear the RDRF flag to 0. However, the RDRF flag is cleared automatically when the DTC is cleared by an RXI interrupt and reads data from RDR.

Legend:
 ∨ : Logical OR

Figure 15.9 Sample Serial Reception Flowchart (1)

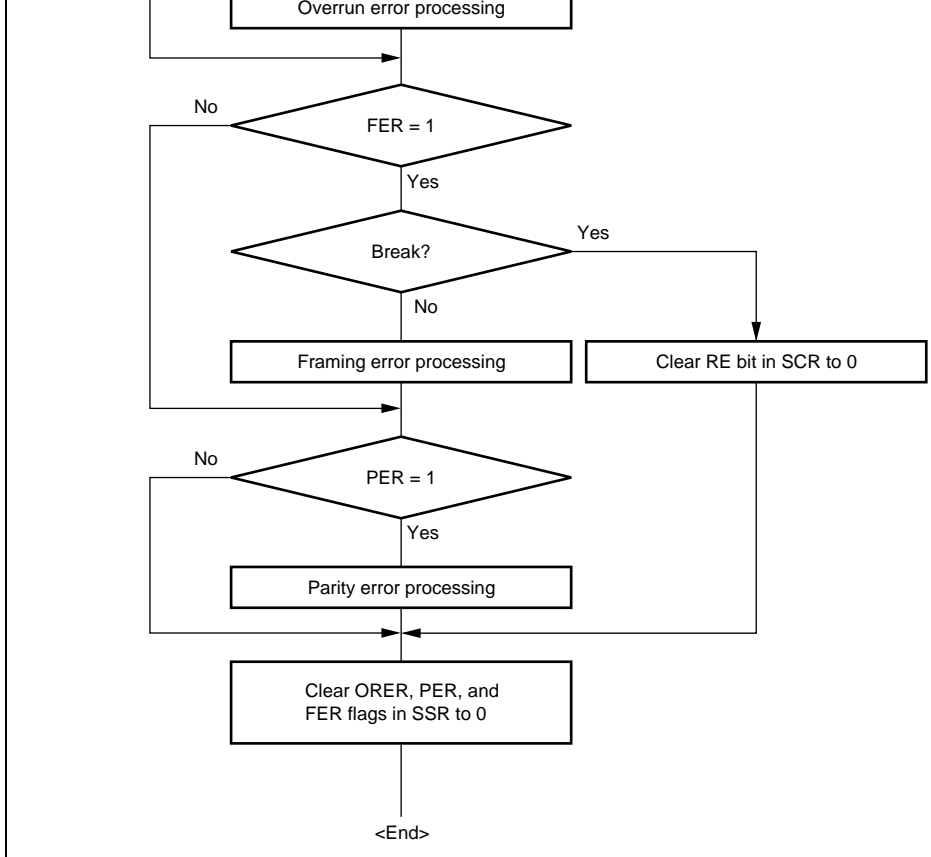
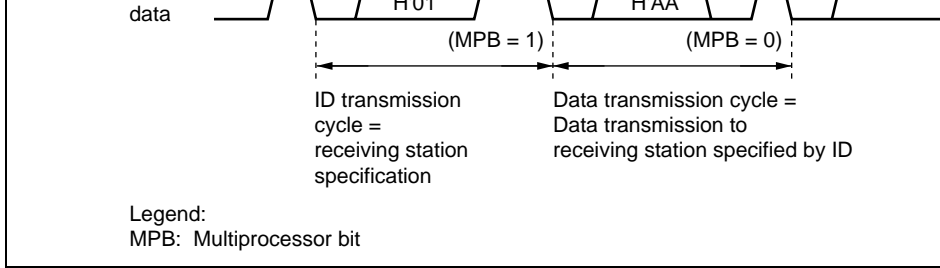


Figure 15.9 Sample Serial Reception Flowchart (2)

the specified receiving station. The multiprocessor bit is used to differentiate between the transmission cycle and the data transmission cycle. If the multiprocessor bit is 1, the cycle is a transmission cycle, and if the multiprocessor bit is 0, the cycle is a data transmission cycle. Figure 15.10 shows an example of inter-processor communication using the multiprocessor format. The transmitting station first sends the ID code of the receiving station with which it wants to communicate in serial communication as data with a 1 multiprocessor bit added. It then sends transmit data with a 0 multiprocessor bit added. The receiving station skips data until data with a 1 multiprocessor bit is sent. When data with a 1 multiprocessor bit is received, the receiving station compares that data with its own ID. The station whose ID matches then receives the data. Stations whose ID does not match continue to skip data until data with a 1 multiprocessor bit is again received.

The SCI uses the MPIE bit in SCR to implement this function. When the MPIE bit is set to 1, the transfer of receive data from RSR to RDR, error flag detection, and setting the SSR status bits RDRF, FER, and ORER in SSR to 1 are prohibited until data with a 1 multiprocessor bit is received. On reception of a receive character with a 1 multiprocessor bit, the MPB bit in SCR is set to 1 and the MPIE bit is automatically cleared, thus normal reception is resumed. If the MPIE bit in SCR is set to 1 at this time, an RXI interrupt is generated.

When the multiprocessor format is selected, the parity bit setting is invalid. All other settings are the same as those in normal asynchronous mode. The clock used for multiprocessor communication is the same as that in normal asynchronous mode.



**Figure 15.10 Example of Communication Using Multiprocessor Format
(Transmission of Data H'AA to Receiving Station A)**

15.5.1 Multiprocessor Serial Data Transmission

Figure 15.11 shows a sample flowchart for multiprocessor serial data transmission. For a transmission cycle, set the MPBT bit in SSR to 1 before transmission. For a data transmission cycle, clear the MPBT bit in SSR to 0 before transmission. All other SCI operations are as those in asynchronous mode.

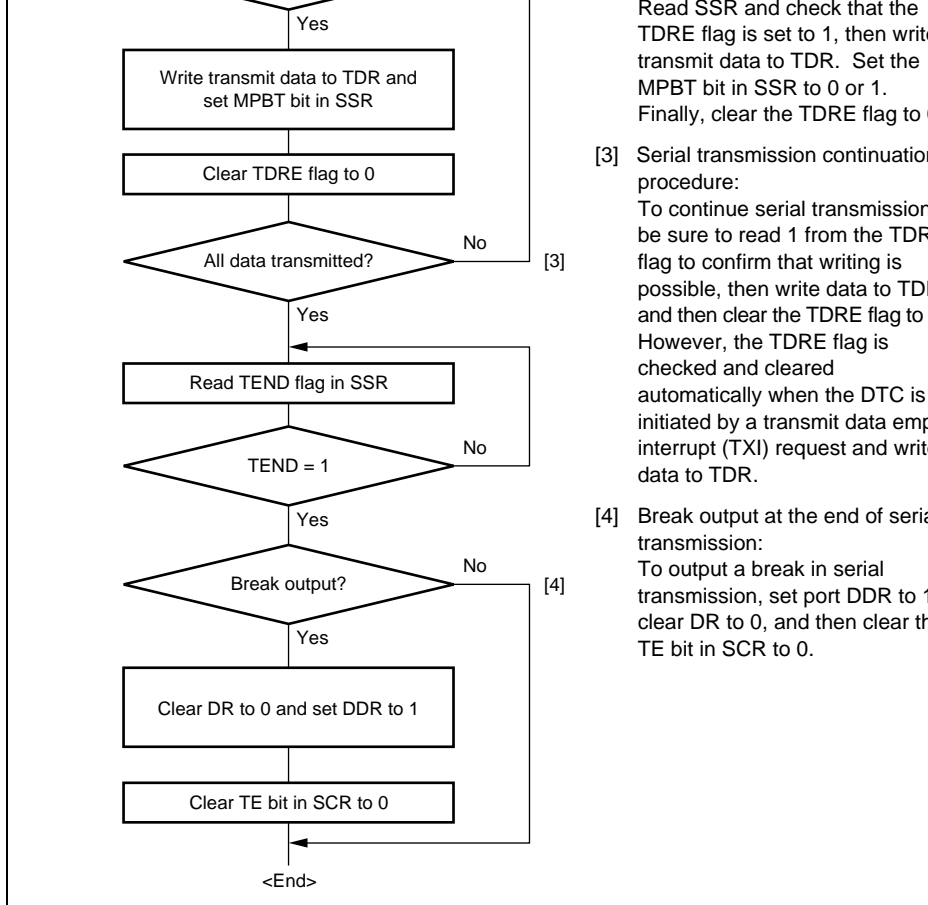
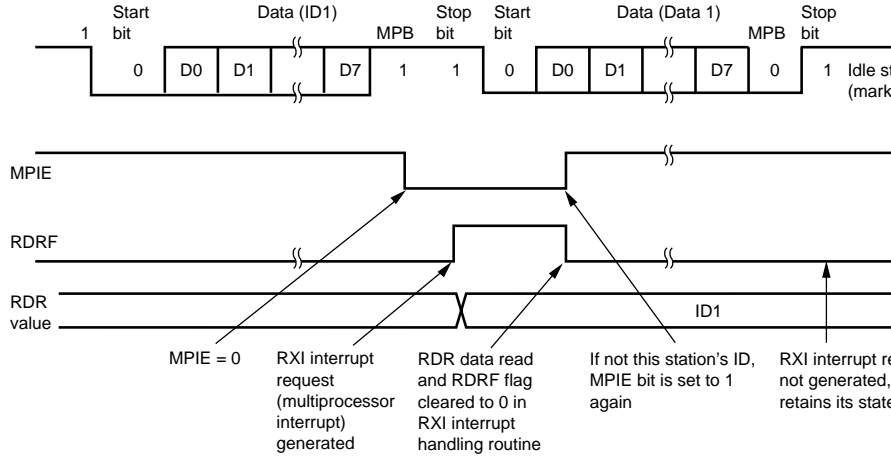
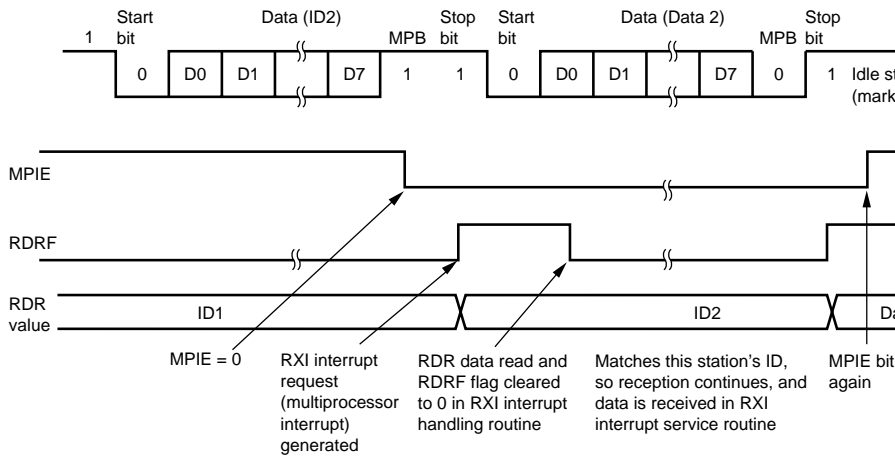


Figure 15.11 Sample Multiprocessor Serial Transmission Flowchart



(a) Data does not match station's ID



(b) Data matches station's ID

**Figure 15.12 Example of SCI Receive Operation
(Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)**

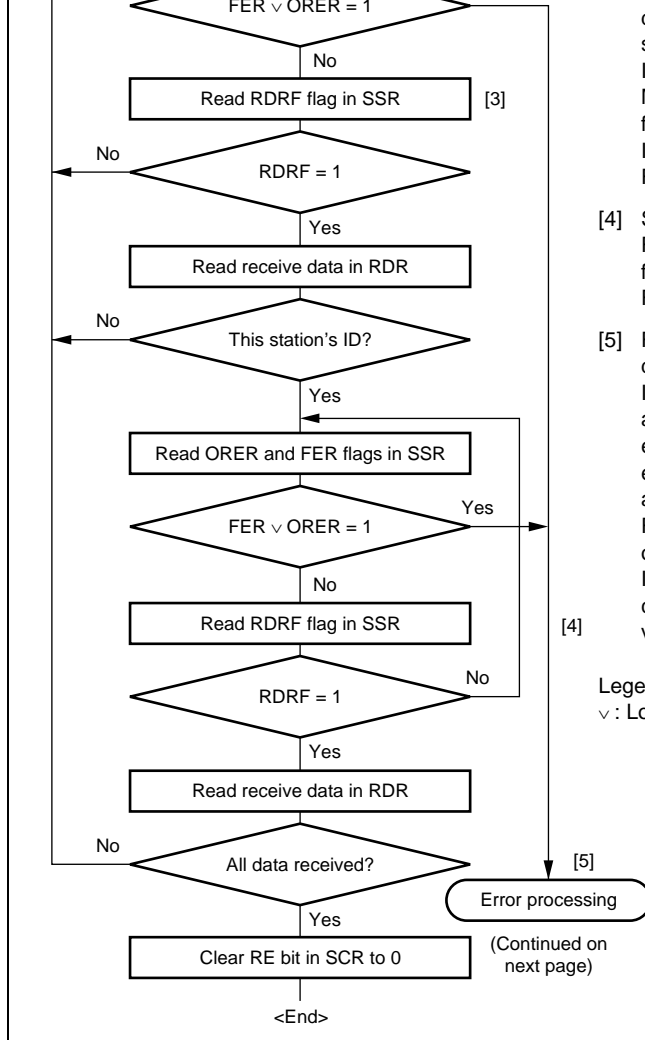


Figure 15.13 Sample Multiprocessor Serial Reception Flowchart (1)

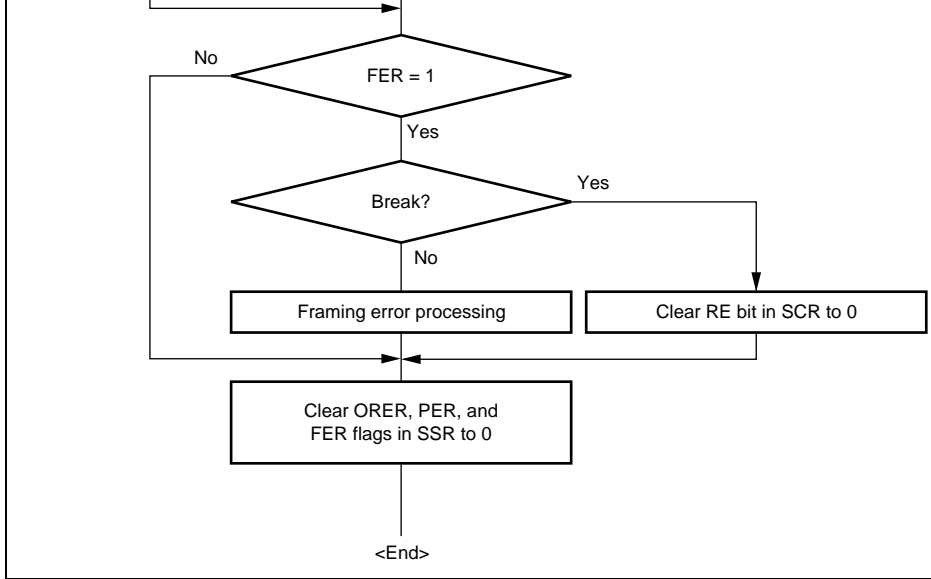


Figure 15.13 Sample Multiprocessor Serial Reception Flowchart (2)

bit is added. Inside the SCI, the transmitter and receiver are independent units, enabling duplex communication by use of a common clock. Both the transmitter and the receiver use a double-buffered structure, so that the next transmit data can be written during transmission and the previous receive data can be read during reception, enabling continuous data transfer.

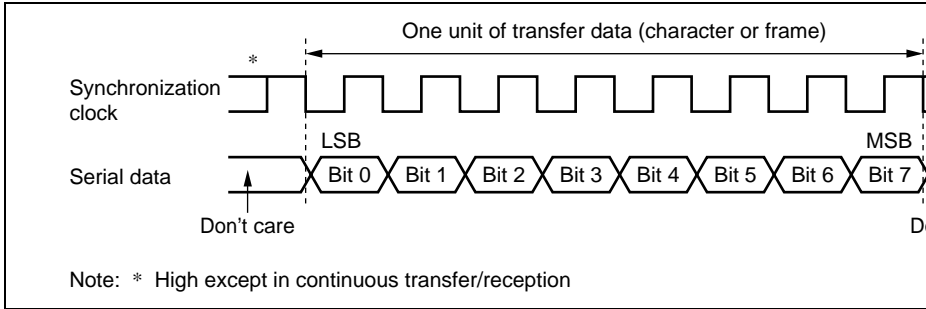


Figure 15.14 Data Format in Clocked Synchronous Communication (LSB)

15.6.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCK pin can be selected, according to the setting of the SCKE0 and CKE0 bits in SCR. When the SCI is operated on an internal clock, the synchronization clock is output from the SCK pin. Eight synchronization clock pulses are output in the transmission of one character, and when no transfer is performed the clock is fixed high.

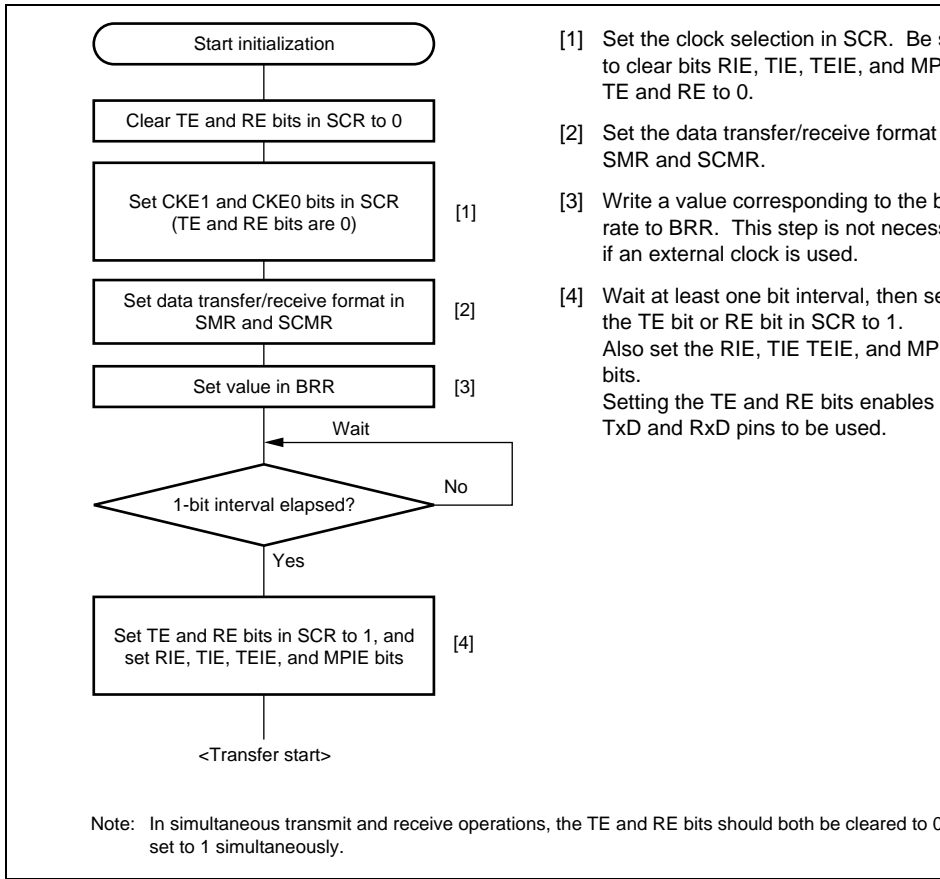


Figure 15.15 Sample SCI Initialization Flowchart

transmission. If the TEIE bit in SCR is set to 1 at this time, a TXI interrupt request is generated. Because the TXI interrupt routine writes the next transmit data to TDR before transmitting the current transmit data has finished, continuous transmission can be enabled.

3. 8-bit data is sent from the TxD pin synchronized with the output clock when output mode has been specified and synchronized with the input clock when use of an external clock has been specified.
4. The SCI checks the TDRE flag at the timing for sending the last bit.
5. If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TxD pin maintains the output state of the last bit. If the TEIE bit in SCR is set to 1 at this time, a TEI interrupt is generated. The SCK pin is fixed high.

Figure 15.17 shows a sample flow chart for serial data transmission. Even if the TDRE flag is cleared to 0, transmission will not start while a receive error flag (ORER, FER, or PER) is set. Make sure to clear the receive error flags to 0 before starting transmission. Note that clearing the TDRE bit to 0 does not clear the receive error flags.

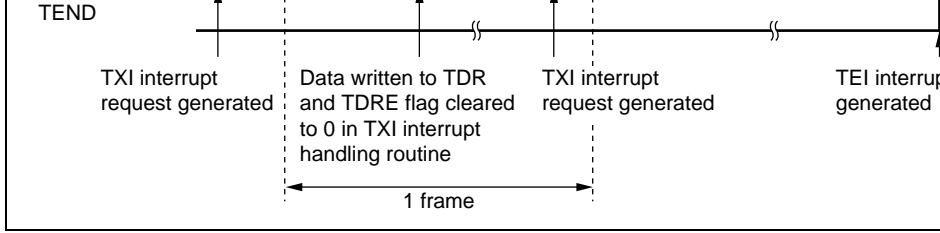
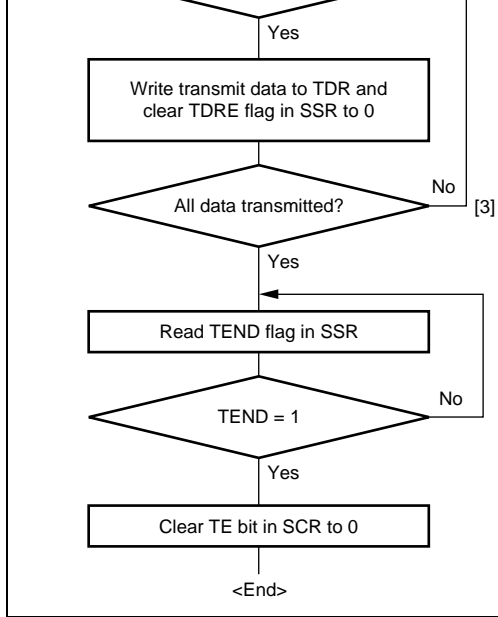


Figure 15.16 Example of SCI Transmit Operation in Clocked Synchronous



[3] Serial transmission continuation procedure:
 To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear TDRE flag to 0.
 However, the TDRE flag is checked and cleared automatically when the DTC is initiated by a transmit data empty interrupt (TXI) request and writes data to TDR.

Figure 15.17 Sample Serial Transmission Flowchart

- set to 1) occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The ORER bit remains to be set to 1.
3. If reception finishes successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Because the RXI interrupt routine reads the receive data transferred to RDR, reception of the next receive data has finished, continuous reception can be enabled.

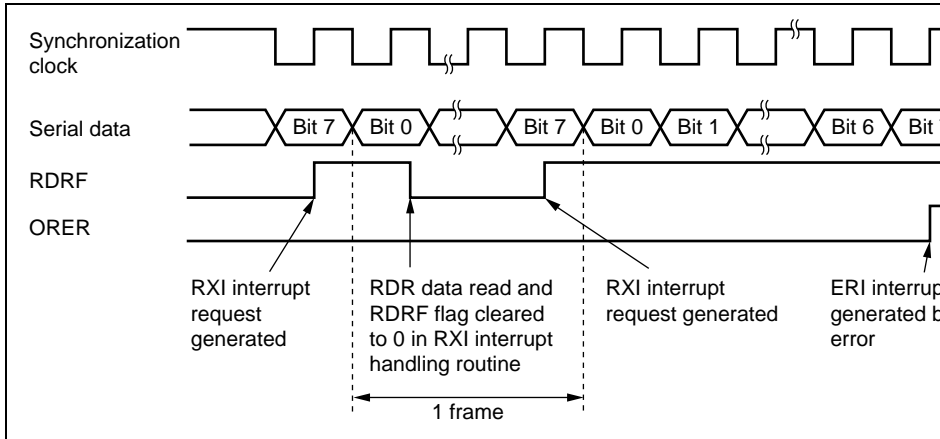
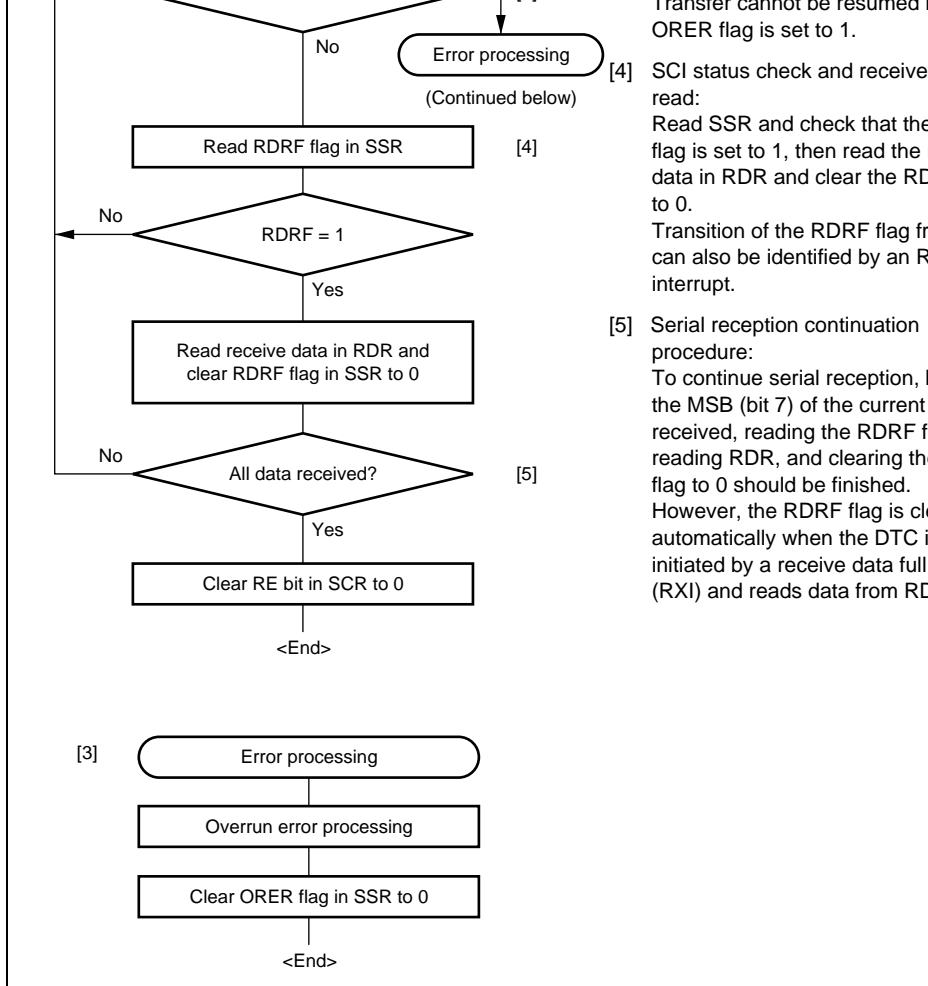


Figure 15.18 Example of SCI Receive Operation in Clocked Synchronous Mode

Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the FER, PER, and RDRF bits to 0 before resuming reception. Figure 15.19 shows a sample timing diagram for serial data reception.



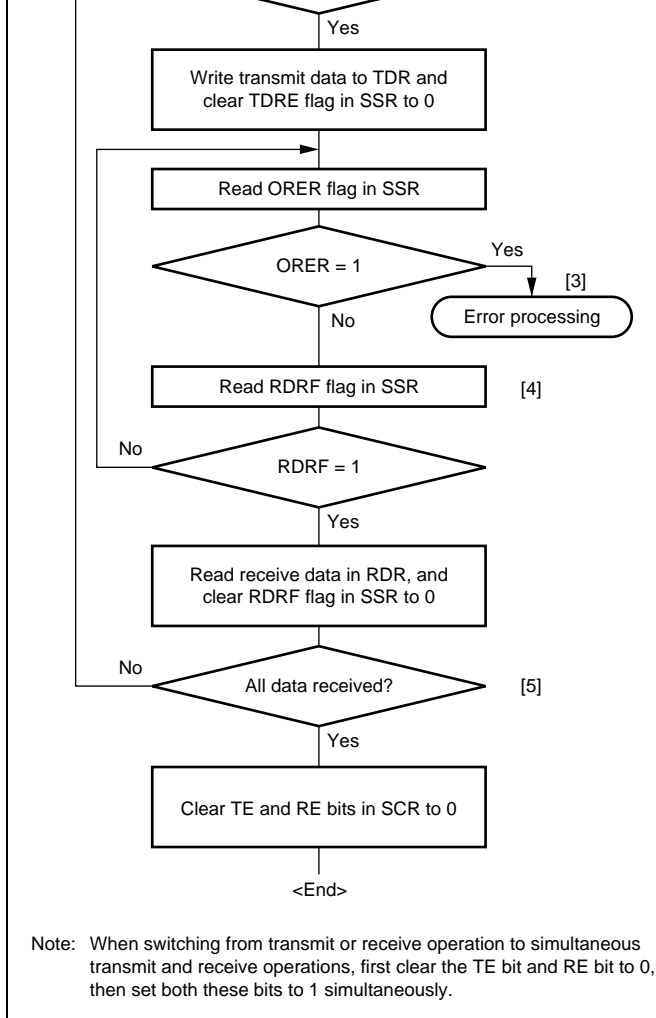
Transfer cannot be resumed. ORER flag is set to 1.

[4] SCI status check and receive read:
Read SSR and check that the flag is set to 1, then read the data in RDR and clear the RDR to 0.
Transition of the RDRF flag from 1 to 0 can also be identified by an RDR interrupt.

[5] Serial reception continuation procedure:
To continue serial reception, read the MSB (bit 7) of the current data received, reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0 should be finished. However, the RDRF flag is cleared automatically when the DTC is initiated by a receive data full (RXI) and reads data from RDR.

Figure 15.19 Sample Serial Reception Flowchart

a single instruction. To switch from receive mode to simultaneous transmit and receive, check that the SCI has finished reception, and clear the RE bit to 0. Then after checking RDRF bit in SSR and receive error flags (ORER, FER, and PER) are cleared to 0, set the RE bits to 1 simultaneously with a single instruction.



data to TDR and clear the TDRE flag to 0. Transition of the TDRE flag from 1 to 0 can also be identified by a data full interrupt.

[3] Receive error processing:
If a receive error occurs, the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag. Transmission/reception can be resumed if the ORER flag is cleared.

[4] SCI status check and receive data read:
Read SSR and check that the ORER flag is set to 1, then read the receive data in RDR and clear the TDRE flag to 0. Transition of the TDRE flag from 1 to 0 can also be identified by a data full interrupt.

[5] Serial transmission/reception continuation procedure:
To continue serial transmission or reception, before the MSB of the current frame is received, read the RDRF flag, read the receive data, and clearing the RDRF flag. Before the MSB (bit 7) of the next frame is transmitted, read the TDRE flag to confirm that the transmission is possible. Then write data to TDR and clear the TDRE flag to 0. However, the TDRE flag is cleared automatically when the DTC is initiated by a transmit empty interrupt (TXI) request. Similarly, the RDRF flag is cleared automatically when the DTC is initiated by a receive data full interrupt (RXI) and the receive data from RDR.

Figure 15.20 Sample Flowchart of Simultaneous Serial Transmission and Reception

In the system defined by the IrDA standard version 1.0, communication is started at a transfer rate of 9600 bps, which can be modified as required. The IrDA interface provided by this LSI incorporates the capability of automatic modification of the transfer rate; the transfer rate can be modified through programming.

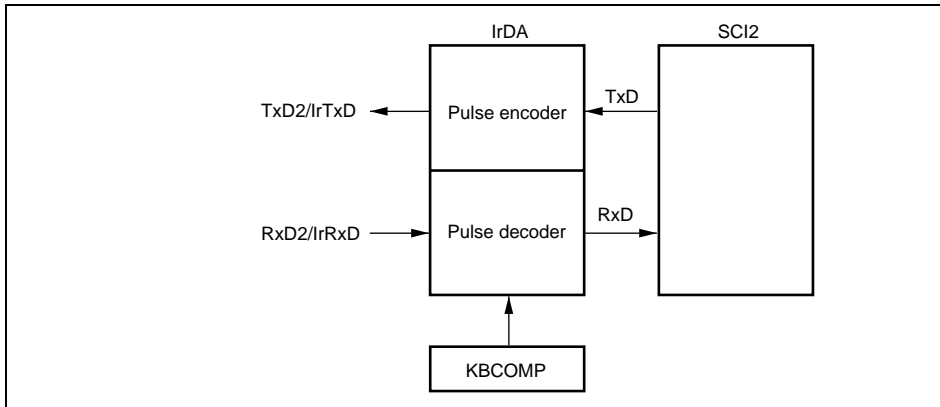


Figure 15.21 IrDA Block Diagram

MHz, a high-level pulse width of at least 1.4 μ s to 1.6 μ s can be specified.

For serial data of level 1, no pulses are output.

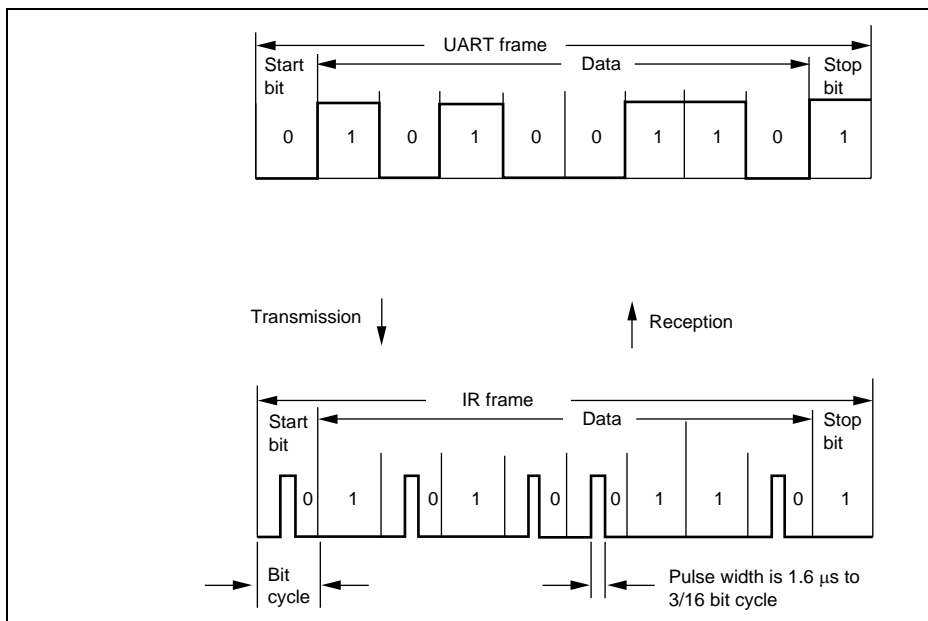


Figure 15.22 IrDA Transmission and Reception

Reception: During reception, IR frames are converted to UART frames using the IrDA before inputting to SCI_2.

Data of level 0 is output each time a high-level pulse is detected and data of level 1 is no pulse is detected in a bit cycle. If a pulse has a high-level width of less than 1.41 μ s minimum width allowed, the pulse is recognized as level 0.

2	010	010	010	010	010	—
2.097152	010	010	010	010	010	—
2.4576	010	010	010	010	010	—
3	011	011	011	011	011	—
3.6864	011	011	011	011	011	01
4.9152	011	011	011	011	011	01
5	011	011	011	011	011	01
6	100	100	100	100	100	10
6.144	100	100	100	100	100	10
7.3728	100	100	100	100	100	10
8	100	100	100	100	100	10
9.8304	100	100	100	100	100	10
10	100	100	100	100	100	10
12	101	101	101	101	101	10
12.288	101	101	101	101	101	10
14	101	101	101	101	101	10
14.7456	101	101	101	101	101	10
16	101	101	101	101	101	10
16.9344	101	101	101	101	101	10
17.2032	101	101	101	101	101	10
18	101	101	101	101	101	10
19.6608	101	101	101	101	101	10
20	101	101	101	101	101	10

Legend:

—: An SCI bit rate setting cannot be made.



When the RDRF flag in SSR is set to 1, an RXI interrupt request is generated. When the TDRE, FER, or PER flag in SSR is set to 1, an ERI interrupt request is generated. An RXI interrupt request activates the DTC to allow data transfer. The RDRF flag is automatically cleared to 0 after data transfer by the DTC.

A TEI interrupt is requested when the TEND flag is set to 1 while the TEIE bit is set to 1. If an ERI interrupt and a TXI interrupt are requested simultaneously, the TXI interrupt has priority over the ERI interrupt acceptance. However, note that if the TDRE and TEND flags are cleared simultaneously, the TXI interrupt routine, the SCI cannot branch to the TEI interrupt routine later.

Table 15.11 SCI Interrupt Sources

Channel	Name	Interrupt Source	Interrupt Flag	DTC Activation
0	ERI0	Receive error	ORER, FER, PER	Not possible
	RXI0	Receive data full	RDRF	Possible
	TXI0	Transmit data empty	TDRE	Possible
	TEI0	Transmit end	TEND	Not possible
1	ERI1	Receive error	ORER, FER, PER	Not possible
	RXI1	Receive data full	RDRF	Possible
	TXI1	Transmit data empty	TDRE	Possible
	TEI1	Transmit end	TEND	Not possible
2	ERI2	Receive error	ORER, FER, PER	Not possible
	RXI2	Receive data full	RDRF	Possible
	TXI2	Transmit data empty	TDRE	Possible
	TEI2	Transmit end	TEND	Not possible

15.9.2 Break Detection and Processing

When framing error detection is performed, a break can be detected by reading the RxDR directly. In a break, the input from the RxD pin becomes all 0s, and so the FER flag in SSR is set to 1, and the PER flag may also be set. Note that, since the SCI continues the receive operation after receiving a break, even if the FER flag is cleared to 0, it will be set to 1 again.

15.9.3 Mark State and Break Detection

When the TE bit in SCR is 0, the TxD pin is used as an I/O port whose direction (input/output) and level are determined by DR and DDR of the port. This can be used to set the TxD pin to mark state (high level) or send a break during serial data transmission. To maintain the communication line at mark state until TE is set to 1, set both DDR and DR to 1. Since TE is cleared to 0 at this point, the TxD pin becomes an I/O port, and 1 is output from the TxD pin. To send a break during serial transmission, first set DDR to 1 and DR to 0, and then clear TE to 0. When the TE bit is cleared to 0, the transmitter is initialized regardless of the current transmission state, the TxD pin becomes an I/O port, and 0 is output from the TxD pin.

15.9.4 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode)

Transmission cannot be started when a receive error flag (ORER, FER, or RER) is set in SSR, even if the TDRE flag in SSR is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be cleared to 0 even if the TE bit in SCR is cleared to 0.

15.9.5 Relation between Writing to TDR and TDRE Flag

Data can be written to TDR irrespective of the TDRE flag status in SSR. However, if data is written to TDR when the TDRE flag is 0, that is, when the previous data has not been transferred to TSR yet, the previous data in TDR is lost. Be sure to write transmit data to TDR after verifying that the TDRE flag is set to 1.

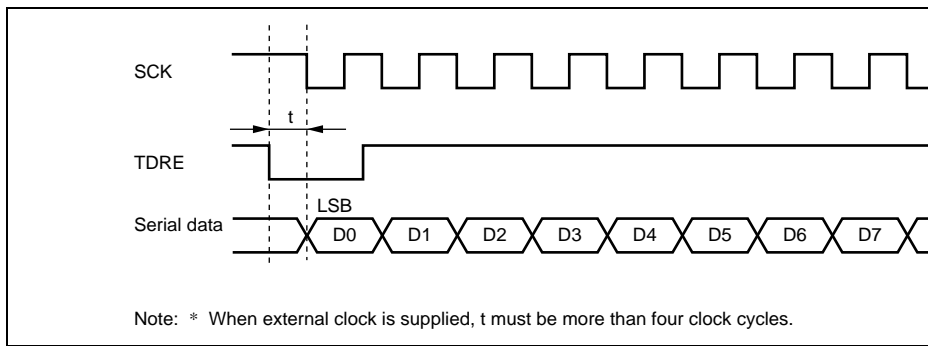


Figure 15.23 Example of Transmission Using DTC in Clocked Synchronous Mode

15.9.7 SCI Operations during Mode Transitions

Transmission: Before making a transition to module stop, software standby, or sub-sleep mode, stop all transmit operations ($TE = TIE = TEIE = 0$). TSR, TDR, and SSR are reset. The output pins during each mode depend on the port settings, and the pins output a high signal after mode cancellation. If a transition is made during data transmission, the data transmitted will be undefined.

To transmit data in the same transmission mode after mode cancellation, set TE to 1, write to TDR, clear TDRE in this order, and then start transmission. To transmit data in a different transmission mode, initialize the SCI first.

Figure 15.24 shows a sample flowchart for mode transition during transmission. Figure 15.26 show the pin states during transmission.

Before making a transition from the transmission mode using DTC transfer to module stop, software standby, or sub-sleep mode, stop all transmit operations ($TE = TIE = TEIE = 0$). TE and TIE to 1 after mode cancellation generates a TXI interrupt request to start transmission using the DTC.

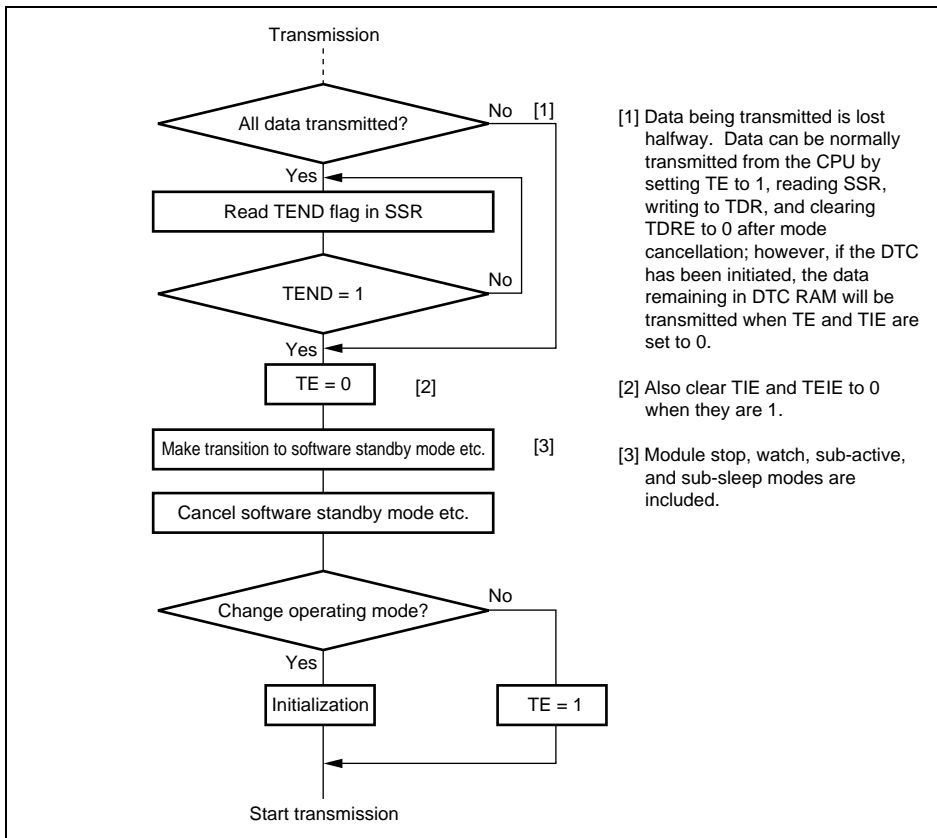


Figure 15.24 Sample Flowchart for Mode Transition during Transmission

Figure 15.25 Pin States during Transmission in Asynchronous Mode (Internal Clock)

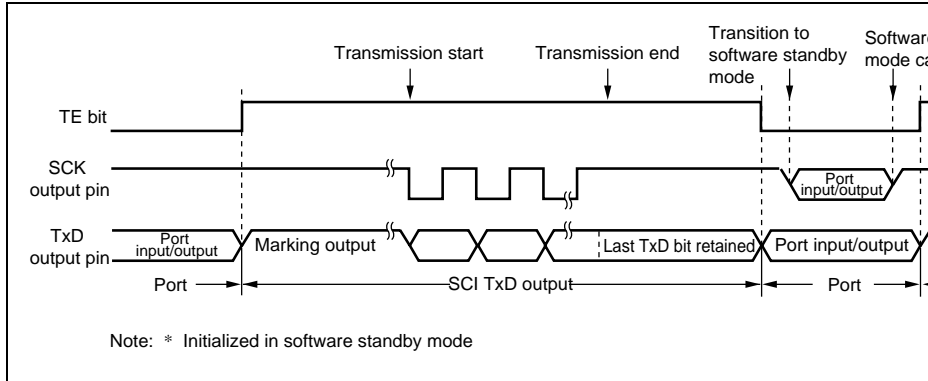


Figure 15.26 Pin States during Transmission in Clocked Synchronous Mode (Internal Clock)

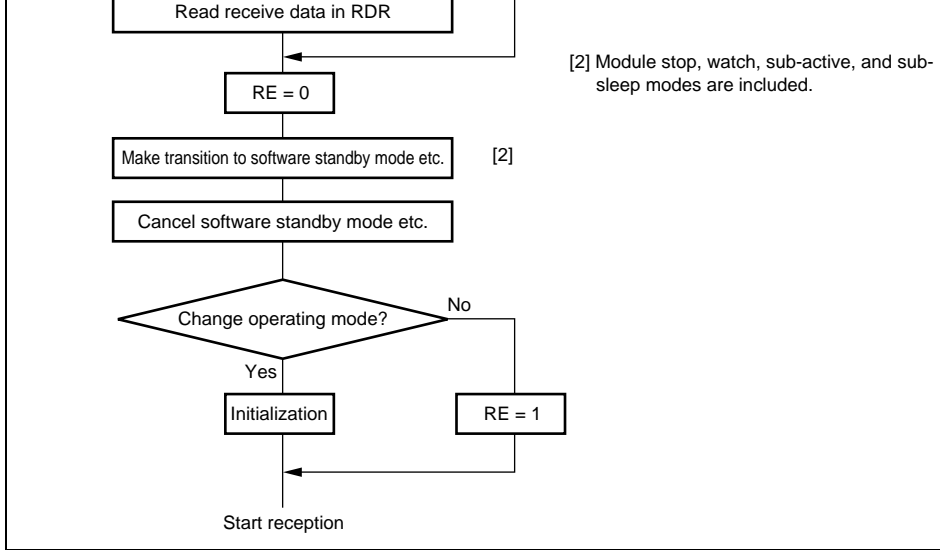


Figure 15.27 Sample Flowchart for Mode Transition during Reception

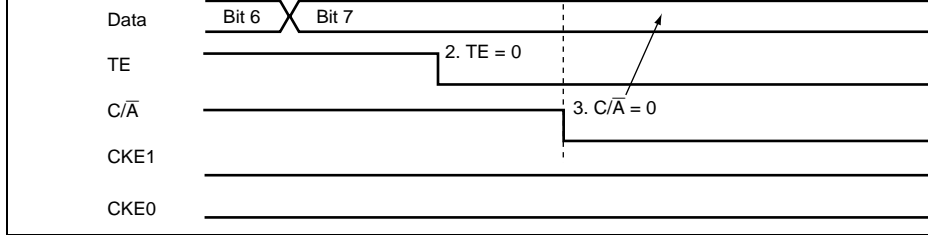


Figure 15.28 Switching from SCK Pins to Port Pins

To prevent the low pulse output that is generated when switching the SCK pins to the port pins, specify the SCK pins for input (pull up the SCK/port pins externally), and follow the procedure below with $DDR = 1$, $DR = 1$, $C/\bar{A} = 1$, $CKE1 = 0$, $CKE1 = 0$, and $TE = 1$.

1. End serial data transmission
2. TE bit = 0
3. CKE1 bit = 1
4. C/\bar{A} bit = 0 (switch to port output)
5. CKE1 bit = 0

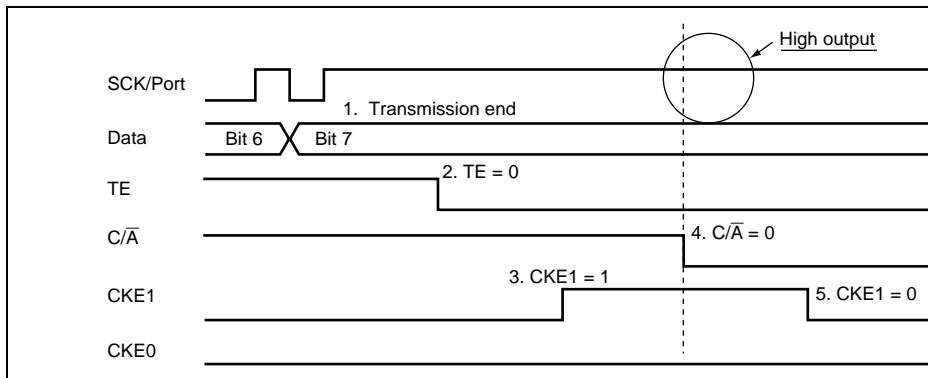


Figure 15.29 Prevention of Low Pulse Output at Switching from SCK Pins to Port Pins

This LSI has a two-channel I²C bus interface. The I²C bus interface conforms to and provides a subset of the Philips I²C bus (inter-IC bus) interface functions. The register configuration for the I²C bus differs partly from the Philips configuration, however.

16.1 Features

- Selection of addressing format or non-addressing format
 - I²C bus format: addressing format with an acknowledge bit, for master/slave operation
 - Clocked synchronous serial format: non-addressing format without an acknowledge bit, for master operation only
 - Formatless (for IIC_0 only): non-addressing format with a clock pin dedicated to the master; formatless; for slave operation only
- Conforms to Philips I²C bus interface (I²C bus format)
- Two ways of setting slave address (I²C bus format)
- Start and stop conditions generated automatically in master mode (I²C bus format)
- Selection of the acknowledge output level in reception (I²C bus format)
- Automatic loading of an acknowledge bit in transmission (I²C bus format)
- Wait function in master mode (I²C bus format)
 - A wait can be inserted by driving the SCL pin low after data transfer, excluding the acknowledge.
 - The wait can be cleared by clearing the interrupt flag.
- Wait function (I²C bus format)
 - A wait request can be generated by driving the SCL pin low after data transfer.
 - The wait request is cleared when the next transfer becomes possible.
- Interrupt sources
 - Data transfer end (including when a transition to transmit mode with I²C bus format occurs, when ICDR data is transferred, or during a wait state)

- Four pins: I²C/SDA0, I²C/SDA1, I²C/SDA2, and I²C/SDA3 (normally I²C pull outputs) function as NMOS open-drain outputs when the bus drive function is selected.
- Automatic switching from formatless mode to I²C bus format (IIC_0 only)
 - Formatless operation (no start/stop conditions, non-addressing mode) in slave mode
 - Operation using a common data pin (SDA) and independent clock pins (VSYN0 and VSYN1)
 - Automatic switching from formatless mode to I²C bus format on the fall of the SDA pin

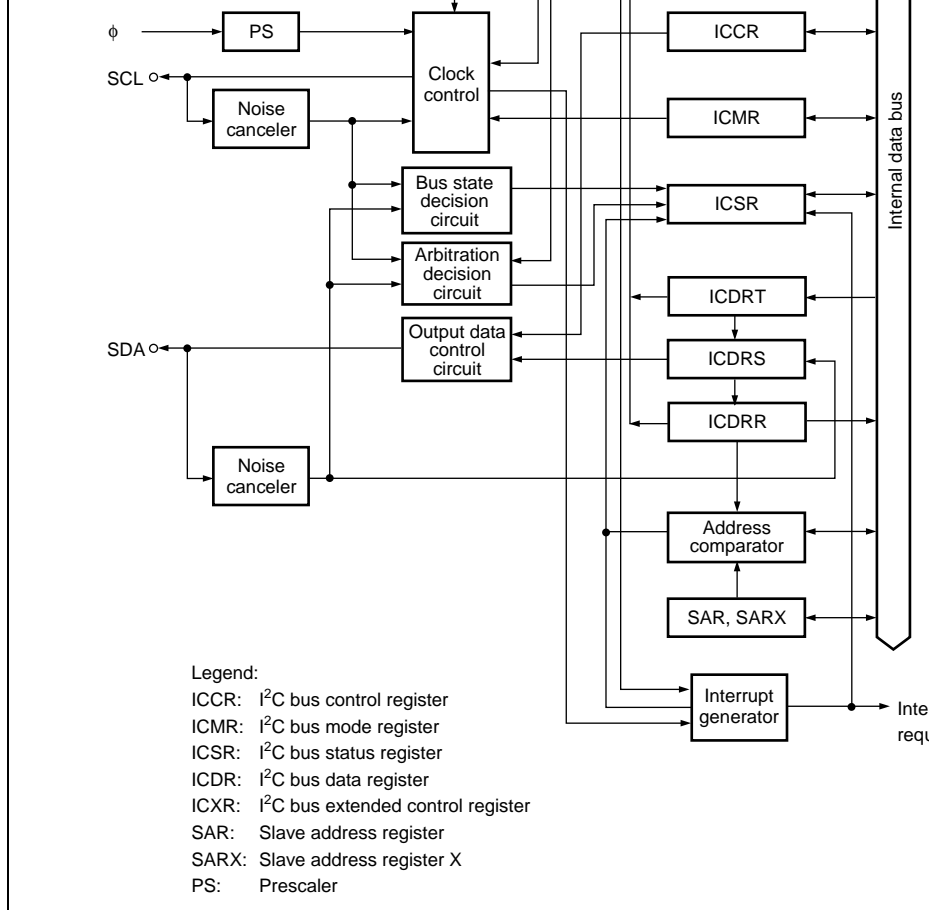


Figure 16.1 Block Diagram of I²C Bus Interface

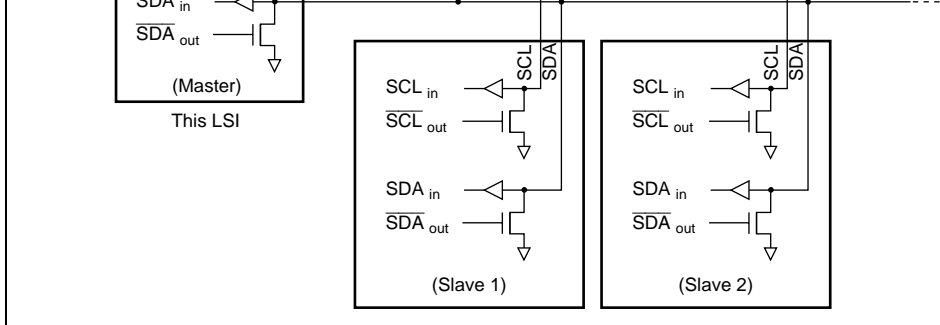


Figure 16.2 I²C Bus Interface Connections (Example: This LSI as Master)

16.2 Input/Output Pins

Table 16.1 summarizes the input/output pins used by the I²C bus interface.

Table 16.1 Pin Configuration

Channel	Symbol*	Input/Output	Function
0	SCL0	Input/Output	Serial clock input/output pin of IIC
	SDA0	Input/Output	Serial data input/output pin of IIC
	VSYNCl	Input	Formatless serial clock input pin of IIC
1	SCL1	Input/Output	Serial clock input/output pin of IIC
	SDA1	Input/Output	Serial data input/output pin of IIC

Note: * In the text, the channel subscript is omitted, and only SCL and SDA are used.

- I²C bus status register (ICSR)
- I²C bus data register (ICDR)
- I²C bus mode register (ICMR)
- Slave address register (SAR)
- Second slave address register (SARX)
- I²C bus extended control register (ICXR)
- DDC switch register (DDCSWR) (for IIC_0 only)

16.3.1 I²C Bus Data Register (ICDR)

ICDR is an 8-bit readable/writable register that is used as a transmit data register when transmitting and a receive data register when receiving. ICDR is internally divided into transmit register (ICDRS), receive buffer (ICDRR), and transmit buffer (ICDRT). Data transfers to these three registers are performed automatically in accordance with changes in the bus that they affect the status of internal flags such as ICDRE and ICDRF.

In master transmit mode with the I²C bus format, writing transmit data to ICDR should be performed after start condition detection. When the start condition is detected, previous data is ignored. In slave transmit mode, writing should be performed after the slave address is detected and the TRS bit is automatically changed to 1.

If the IIC is in transmit mode (TRS = 1) and ICDRT has the next transmit data (the ICDRT bit is 0) after successful transmission/reception of one frame of data using ICDRS, data is transferred automatically from ICDRT to ICDRS.

If the IIC is in transmit mode (TRS = 1) and ICDRT has the next data (the ICDRE flag is 1) is transferred automatically from ICDRT to ICDRS, following transmission of one frame of data using ICDRS. When the ICDRE flag is 1 and the next transmit data writing is waited, data is transferred automatically from ICDRT to ICDRS by writing to ICDR. If I²C is in receive mode (TRS = 0), no data is transferred from ICDRT to ICDRS. Note that data should not be written to ICDR in receive mode.

and receive data are stored differently. Transmit data should be written justified toward the MSB side when MLS = 0 in ICMR, and toward the LSB side when MLS = 1. Receive data should be read from the LSB side when MLS = 0, and from the MSB side when MLS = 1.

ICDR can be written to and read from only when the ICE bit is set to 1 in ICCR. The initial value of ICDR is undefined.

16.3.2 Slave Address Register (SAR)

SAR sets the slave address and selects the communication format. If the LSI is in slave mode and the I²C bus format selected, when the FS bit is set to 0 and the upper 7 bits of SAR match the upper 7 bits of the first frame received after a start condition, the LSI operates as the slave device specified by the master device. SAR can be accessed only when the ICE bit in ICCR is set to 0.

Bit	Bit Name	Initial Value	R/W	Description
7	SVA6	0	R/W	Slave Address 6 to 0
6	SVA5	0	R/W	Set a slave address.
5	SVA4	0	R/W	
4	SVA3	0	R/W	
3	SVA2	0	R/W	
2	SVA1	0	R/W	
1	SVA0	0	R/W	
0	FS	0	R/W	Format Select Selects the communication format together with the SW bit in SARX and the SW bit in DDCSWR. Refer to Section 16.2. This bit should be set to 0 when general call address recognition is performed.

Bit	Bit Name	Initial Value	R/W	Description
7	SVAX6	0	R/W	Second Slave Address 6 to 0
6	SVAX5	0	R/W	Set the second slave address.
5	SVAX4	0	R/W	
4	SVAX3	0	R/W	
3	SVAX2	0	R/W	
2	SVAX1	0	R/W	
1	SVAX0	0	R/W	
0	FSX	1	R/W	Format Select X Selects the communication format together with the SW bit in SAR and the SW bit in DDCSWR. Refer to

			<ul style="list-style-type: none"> • SAR slave address recognized • SARX slave address ignored • General call address recognized
1	0		I ² C bus format <ul style="list-style-type: none"> • SAR slave address ignored • SARX slave address recognized • General call address ignored
		1	Clocked synchronous serial format <ul style="list-style-type: none"> • SAR and SARX slave addresses ignored • General call address ignored
1	0	0	Formatless mode (start/stop conditions not detected) <ul style="list-style-type: none"> • Acknowledge bit used
	1	0	
		1	Formatless mode (start/stop conditions not detected) <ul style="list-style-type: none"> • No acknowledge bit Do not set this mode when automatic switching to I ² C bus format is performed by means of the DDCSW.

- I²C bus format: addressing format with an acknowledge bit
- Clocked synchronous serial format: non-addressing format without an acknowledge bit (master mode only)
- Formatless mode (for IIC_0 only): non-addressing format with or without an acknowledge bit (slave mode only, start/stop conditions not detected)

1: LSB-first

Set this bit to 0 when the I²C bus format is used.

6	WAIT	0	R/W	<p>Wait Insertion Bit</p> <p>This bit is valid only in master mode with the I²C bus format.</p> <p>0: Data and the acknowledge bit are transferred consecutively with no wait inserted.</p> <p>1: After the fall of the clock for the final data byte, the IRIC flag is set to 1 in ICCR, and a wait is inserted (with SCL at the low level). When the IRIC flag is cleared to 0 in ICCR, the wait ends and the acknowledge bit is transferred.</p> <p>For details, refer to section 16.4.7, IRIC Setting and SCL Control.</p>
5	CKS2	0	R/W	Transfer Clock Select 2 to 0
4	CKS1	0	R/W	These bits are used only in master mode.
3	CKS0	0	R/W	These bits select the required transfer rate, together with the IICX1 (IIC_1) and IICX0 (IIC_0) bits in STCR. For details, refer to table 16.3.

transfer.

I ² C Bus Format	Clocked Synchronous Serial
000: 9 bits	000: 8 bits
001: 2 bits	001: 1 bits
010: 3 bits	010: 2 bits
011: 4 bits	011: 3 bits
100: 5 bits	100: 4 bits
101: 6 bits	101: 5 bits
110: 7 bits	110: 6 bits
111: 8 bits	111: 7 bits

0	0	1	0	φ/48	104 kHz	167 kHz	208 kHz	333 kHz
0	0	1	1	φ/64	78.1 kHz	125 kHz	156 kHz	250 kHz
0	1	0	0	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz
0	1	0	1	φ/100	50.0 kHz	80.0 kHz	100 kHz	160 kHz
0	1	1	0	φ/112	44.6 kHz	71.4 kHz	89.3 kHz	143 kHz
0	1	1	1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz
1	0	0	0	φ/56	89.3 kHz	143 kHz	179 kHz	286 kHz
1	0	0	1	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz
1	0	1	0	φ/96	52.1 kHz	83.3 kHz	104 kHz	167 kHz
1	0	1	1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz
1	1	0	0	φ/160	31.3 kHz	50.0 kHz	62.5 kHz	100 kHz
1	1	0	1	φ/200	25.0 kHz	40.0 kHz	50.0 kHz	80.0 kHz
1	1	1	0	φ/224	22.3 kHz	35.7 kHz	44.6 kHz	71.4 kHz
1	1	1	1	φ/256	19.5 kHz	31.3 kHz	39.1 kHz	62.5 kHz

Note: * Outside the I²C bus interface specifications (standard mode: max. 100 kHz; mode: max. 400 kHz)

1: I²C bus interface modules can perform transmit operation, and the ports function as the SCL input/output pins. ICMR and ICDR can be accessed.

6	IEIC	0	R/W	<p>I²C Bus Interface Interrupt Enable</p> <p>0: Disables interrupts from the I²C bus interface CPU</p> <p>1: Enables interrupts from the I²C bus interface CPU.</p>
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	<p>Transmit/Receive Select</p> <p>00: Slave receive mode</p> <p>01: Slave transmit mode</p> <p>10: Master receive mode</p> <p>11: Master transmit mode</p> <p>Both these bits will be cleared by hardware when a master loses in a bus contention in master mode with the 7-bit format. In slave receive mode with I²C bus format, the R/W bit in the first frame immediately after the start condition sets these bits in receive mode or transmit mode automatically by hardware.</p> <p>Modification of the TRS bit during transfer is disabled. After transfer is completed, and the changeover is made, the TRS bit is set to 00 on completion of the transfer.</p>

- condition 1)
- 2. When 1 is written in MST after reading MST clearing condition 2)

[TRS clearing conditions]

- 1. When 0 is written by software (except for TRS clearing condition 3)
- 2. When 0 is written in TRS after reading TRS clearing condition 3)
- 3. When lost in bus contention in I²C bus format slave mode
- 4. When the SW bit in DDCCSWR is changed

[TRS setting conditions]

- 1. When 1 is written by software (except for TRS setting conditions 3 and 4)
- 2. When 1 is written in TRS after reading TRS clearing conditions 3 and 4)
- 3. When 1 is received as the R/W bit after the address matching in I²C bus format slave mode

3	ACKE	0	R/W	<p>Acknowledge Bit Decision and Selection</p> <p>0: The value of the acknowledge bit is ignored. Continuous transfer is performed. The value of the received acknowledge bit is not indicated in ICSR, which is always 0.</p> <p>1: If the received acknowledge bit is 1, continuous transfer is halted.</p> <p>Depending on the receiving device, the acknowledge bit may be significant, in indicating completion of reception of the received data, for instance, or may be insignificant and have no significance.</p>
---	------	---	-----	--



In slave mode:

- Writing to the BBSY flag is disabled.

[BBSY setting condition]

When the SDA level changes from high to low under the condition of SCL = high, assuming that the start condition has been issued.

[BBSY clearing condition]

When the SDA level changes from low to high under the condition of SCL = high, assuming that the stop condition has been issued.

To issue a start/stop condition, use the MOV instruction.

The I²C bus interface must be set in master transmit mode before the issue of a start condition. Set MST to 1 before writing 1 in BBSY and 0 in SCP.

The BBSY flag can be read to check whether the bus (SCL, SDA) is busy or free.

The SCP bit is always read as 1. If 0 is written, the value is not stored.

[Setting conditions]

I²C bus format master mode:

- When a start condition is detected in the bus (when the START bit in ICSR is set to 1 because of first frame transmission)
- When a wait is inserted between the data and the acknowledge bit when the WAIT bit is 1 (falling edge of transmit/receive clock)
- At the end of data transfer (rise of the 9th transmit/receive clock while no wait is inserted)
- When a slave address is received after bus arbitration is lost (the first frame after the start condition)
- If 1 is received as the acknowledge bit (while the ACK bit in ICSR is set to 1) when the ACKE bit is 1
- When the AL flag is set to 1 after bus arbitration is lost while the ALIE bit is 1

I²C bus format slave mode:

- When the slave address (SVA or SVAX) matches the master address (when the AAS or AASX flag in ICSR is set to 1) up to the end of data transfer up to the subsequent retransmission start condition or stop condition detection (rise of the 9th transmit/receive clock)
- When the general call address is detected (when the R \bar{W} bit and the ADZ flag in ICSR are set to 1) and at the end of data reception up to the subsequent retransmission start condition or stop condition detection (rise of the 9th receive clock)
- If 1 is received as the acknowledge bit (while the ACK bit in ICSR is set to 1) while the ACKE bit is 1
- When a stop condition is detected (when the STOP bit in ICSR is set to 1) while the STP flag in ICSR is set to 1

When the ICDRE or ICDRF flag is set to 1 in an transmit mode:

- When a start condition is detected in transmit mode (when a start condition is detected in transmit mode and the ICDRE flag is set to 1)
- When data is transferred among the ICDR receive and transmit buffers (when data is transferred from ICDRT to ICDR in transmit mode and the ICDRE flag is set to 1 or when data is transferred from ICDRS to ICDR in receive mode and the ICDRF flag is set to 1)

[Clearing conditions]

- When 0 is written in IRIC after reading IRIC
- When ICDR is read from or written to by the CPU (this may not function as a clearing condition depending on the situation. For details, see the description of the ICDR operation given below.)

Note: * Only 0 can be written, to clear the flag.

When the DTC is used, IRIC is cleared automatically and transfer can be performed continuously without CPU intervention.

When, with the I²C bus format selected, IRIC is set to 1 and an interrupt is generated, the source of the interrupt must be checked in order to identify the source that set IRIC to 1. Although each source has its own corresponding flag, caution is needed at the end of a transfer.

When the ICDRE or ICDRF flag is set, the IRTR flag may or may not be set. The IRTR flag (the ICDR DTC start request flag) is not set at the end of a data transfer up to detection of a retransmission start condition or stop condition after a slave address (SVA) or general call address match in the I²C bus format slave mode.

Even when the IRIC flag and IRTR flag are set, the ICDRE or ICDRF flag may not be cleared. IRIC and IRTR flags are not cleared at the end of the specified number of transfers in continuous transfer mode.

1	1	1↑	0	0	1↑	0	0	0	0	0	—	1↑	Start cond
1	—	1	0	0	—	0	0	0	0	—	—	—	Wait state
1	1	1	0	0	—	0	0	0	0	1↑	—	—	Transmiss (ACKE=1)
1	1	1	0	0	1↑	0	0	0	0	0	—	1↑	Transmiss ICDRE=0
1	1	1	0	0	—	0	0	0	0	0	—	0↓	ICDR write state
1	1	1	0	0	—	0	0	0	0	0	—	1	Transmiss ICDRE=1
1	1	1	0	0	—	0	0	0	0	0	—	0↓	ICDR write state or af condition
1	1	1	0	0	1↑	0	0	0	0	0	—	1↑	Automatic from ICDR the above
1	0	1	0	0	1↑	0	0	0	0	—	1↑	—	Reception ICDRF=0
1	0	1	0	0	—	0	0	0	0	—	0↓	—	ICDR read state
1	0	1	0	0	—	0	0	0	0	—	1	—	Reception ICDRF=1
1	0	1	0	0	—	0	0	0	0	—	0↓	—	ICDR read state
1	0	1	0	0	1↑	0	0	0	0	—	1↑	—	Automatic from ICDR with the ab
0↓	0↓	1	0	0	—	0	1↑	0	0	—	—	—	Arbitration
1	—	0↓	0	0	—	0	0	0	0	—	—	0↓	Stop cond

Legend:

0: 0-state retained

1: 1-state retained

—: Previous state retained

0↓: Cleared to 0

1↑: Set to 1

0	1↑/0* ¹	1	0	0	1↑	1↑	—	0	0	0	1↑	1	(SARX ≠ H SARS mat frame (SAR
0	1	1	0	0	—	—	—	—	0	1↑	—	—	Transmissi = 1 and AC
0	1	1	0	0	1↑/0* ²	—	—	—	0	0	—	1↑	Transmissi ICDRE = 0
0	1	1	0	0	—	—	0↓	0↓	0	0	—	0↓	ICDR write above state
0	1	1	0	0	—	—	—	—	0	0	—	1	Transmissi ICDRE = 1
0	1	1	0	0	—	—	0↓	0↓	0	0	—	0↓	ICDR write above state
0	1	1	0	0	1↑/0* ²	—	0	0	0	0	—	1↑	Automatic from ICDR with the ab
0	0	1	0	0	1↑/0* ²	—	—	—	—	—	1↑	—	Reception ICDRF=0
0	0	1	0	0	—	—	0↓	0↓	0↓	—	0↓	—	ICDR read state
0	0	1	0	0	—	—	—	—	—	—	1	—	Reception ICDRF = 1
0	0	1	0	0	—	—	0↓	0↓	0↓	—	0↓	—	ICDR read state
0	0	1	0	0	1↑/0* ²	—	0	0	0	—	1↑	—	Automatic from ICDR with the ab
0	—	0↓	1↑/0* ³	0/1↑* ³	—	—	—	—	—	—	—	0↓	Stop condi

Legend:

0: 0-state retained

1: 1-state retained

—: Previous state retained

0↓: Cleared to 0

1↑: Set to 1

Notes: 1. Set to 1 when 1 is received as a R/W bit following an address.

2. Set to 1 when the AASX bit is set to 1.

3. When ESTP=1, STOP is 0, or when STOP=1, ESTP is 0.

When a stop condition is detected during frame

[Clearing conditions]

- When 0 is written in ESTP after reading ES
- When the IRIC flag in ICCR is cleared to 0

6	STOP	0	R/(W)*	Normal Stop Condition Detection Flag
---	------	---	--------	--------------------------------------

This bit is valid in I²C bus format slave mode.

[Setting condition]

When a stop condition is detected after frame completion.

[Clearing conditions]

- When 0 is written in STOP after reading ST
- When the IRIC flag is cleared to 0

[Setting conditions]

I²C bus format slave mode:

- When the ICDRE or ICDRF flag in ICDR is set to 1 when AASX = 1

Master mode or clocked synchronous serial format with I²C bus format, or formatless mode:

- When the ICDRE or ICDRF flag is set to 1

[Clearing conditions]

- When 0 is written after reading IRTR = 1
- When the IRIC flag is cleared to 0 while ICE = 1

4	AASX	0	R/(W)*	Second Slave Address Recognition Flag
---	------	---	--------	---------------------------------------

In I²C bus format slave receive mode, this flag is set to 1 when the first frame following a start condition matches SVAX6 to SVAX0 in SARX.

[Setting condition]

When the second slave address is detected in slave receive mode and FSX = 0 in SARX

[Clearing conditions]

- When 0 is written in AASX after reading AASX = 1
- When a start condition is detected
- In master mode

- If the internal SCL line is high at the fall of master transmit mode

When ALSL = 1

- If the internal SDA and SDA pin disagree a SCL in master transmit mode
- If the SDA pin is driven low by another device the I²C bus interface drives the SDA pin low start condition instruction was executed in master transmit mode

[Clearing conditions]

- When ICDR is written to (transmit mode) or (receive mode)
- When 0 is written in AL after reading AL = 1

2	AAS	0	R/(W)*	<p>Slave Address Recognition Flag</p> <p>In I²C bus format slave receive mode, this flag is set when the first frame following a start condition match to SVA0 in SAR, or if the general call address is detected.</p> <p>[Setting condition]</p> <p>When the slave address or general call address (including a R/W bit is H'00) is detected in slave receive mode and FS = 0 in SAR</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When ICDR is written to (transmit mode) or (receive mode) • When 0 is written in AAS after reading AAS = 1 • In master mode
---	-----	---	--------	---

FSX = 0

[Clearing conditions]

- When ICDR is written to (transmit mode) or (receive mode)
- When 0 is written in ADZ after reading ADZ
- In master mode

If a general call address is detected while FS=1, FSX=0, the ADZ flag is set to 1; however, the general call address is not recognized (AAS flag is not set to 1).

[clearing conditions]

- When 0 is received as the acknowledge bit
 ACKE=1 in transmit mode
- When 0 is written to the ACKE bit

Receive mode:

0: Returns 0 as acknowledge data after data re

1: Returns 1 as acknowledge data after data re

When this bit is read, the value loaded from the
(returned by the receiving device) is read in tra
(when TRS = 1). In reception (when TRS = 0),
by internal software is read.

When this bit is written, acknowledge data that
after receiving is rewritten regardless of the TR
the ICSR register bit is written using bit-manipu
instructions, the acknowledge data should be r
the acknowledge data setting is rewritten by th
reading value.

Write the ACKE bit to 0 to clear the ACKB flag
transmission is ended and a stop condition is i
master mode, or before transmission is ended
released to issue a stop condition by a master

Note: * Only 0 can be written to clear the flag.

				formatless mode to I ² C bus format 1: Enables automatic switching of IIC channel 0 formatless mode to I ² C bus format
6	SW	0	R/W	DDC Mode Switch 0: Uses IIC channel 0 with the I ² C bus format 1: Uses IIC channel 0 in formatless mode [Setting condition] When 1 is written in SW after reading SW = 0 [Clearing conditions] <ul style="list-style-type: none"> • When 0 is written by software • When a falling edge is detected on the SCL pin SWE = 1
5	IE	0	R/W	DDC Mode Switch Interrupt Enable Bit 0: Disables interrupts when automatic format switching is executed 1: Enables interrupts when automatic format switching is executed
4	IF	0	R/(W)*1	DDC Mode Switch Interrupt Flag Indicates an interrupt request to the CPU is generated when automatic format switching is executed for the IIC channel 0. [Setting condition] When a falling edge is detected on the SCL pin SWE = 1 [Clearing condition] When 0 is written in IF after reading IF = 1

0110: IIC_1 internal latch cleared

0111: IIC_0 and IIC_1 internal latches cleared

1---: Invalid setting

When a write operation is performed on these bits, a clear signal is generated for the internal latch circuit of the corresponding module, and the internal state of the module is initialized.

These bits can only be written to; they are always read as 1. Write data to this bit is not retained.

To perform IIC clearance, bits CLR3 to CLR0 must be written to simultaneously using an MOV instruction. Do not use a bit manipulation instruction such as BCLR.

When clearing is required again, all the bits must be written to in accordance with the setting.

If the function of these bits is not used, set all of the CLR3 to CLR0 bits to 1 when writing to DDCCSWR.

-
- Notes: 1. Only 0 can be written, to clear the flag.
2. This bit is always read as 1.

condition is detected in slave mode.

0: Enables IRIC flag setting and interrupt generation when the stop condition is detected (STOP = 1 or E-stop) in slave mode.

1: Disables IRIC flag setting and interrupt generation when the stop condition is detected.

6	HNDS	0	R/W	<p>Handshake Receive Operation Select</p> <p>Enables or disables continuous receive operation mode.</p> <p>0: Enables continuous receive operation</p> <p>1: Disables continuous receive operation</p> <p>When the HNDS bit is cleared to 0, receive operation is performed continuously after data has been received successfully while ICDRF flag is 0.</p> <p>When the HNDS bit is set to 1, SCL is fixed to the high level and the next data transfer is disabled after data has been received successfully while the ICDRF flag is 0. The SCL line is released and next receive operation is enabled after reading the receive data in ICDR.</p>
---	------	---	-----	---

[Setting conditions]

- When data is received successfully and transferred from ICDRS to ICDRR.
- (1) When data is received successfully while ICDRF = 0 (at the rise of the 9th clock pulse).
 - (2) When ICDR is read successfully in receive mode and data was received while ICDRF = 1.

[Clearing conditions]

- When ICDR (ICDRR) is read.
- When 0 is written to the ICE bit.
- When the IIC is internally initialized using the CLR0 bits in DDOSWR.

When ICDRF is set due to the condition (2) above, ICDRF is temporarily cleared to 0 when ICDR (ICDRR) is read. However, since data is transferred from ICDRS to ICDRR immediately, ICDRF is set to 1 again.

Note that ICDR cannot be read successfully in receive mode (TRS = 1) because data is not transferred from ICDRS to ICDRR. Be sure to read data from ICDR in receive mode (TRS = 0).

thus allowing the next data to be written to.

[Setting conditions]

- When the start condition is detected from the state with I²C bus format or serial format.
- When I²C bus mode is switched to formatless SW bit in DDCSWR is set to 1).
- When data is transferred from ICDRT to ICDR⁰
 1. When data transmission completed while ICDRE = 0 (at the rise of the 9th clock pulse).
 2. When data is written to ICDR in transmit mode while ICDRE = 0 (at the completion of data transmission was completed while ICDRE = 0).

[Clearing conditions]

- When data is written to ICDR (ICDRT).
- When the stop condition is detected with I²C bus format or serial format.
- When 0 is written to the ICE bit.
- When the IIC is internally initialized using the CLR0 bits in DDCSWR.

Note that if the ACKE bit is set to 1 with I²C bus format, enabling acknowledge bit decision, ICDRE is not cleared to 0 when data transmission is completed while the acknowledge bit is 1.

When ICDRE is set due to the condition (2) above, ICDRE is temporarily cleared to 0 when data is written to ICDR (ICDRT); however, since data is transferred from ICDRT to ICDR immediately, ICDRE is set to 1 again. Data is not transferred from ICDR to ICDRT when TRS = 0 because the ICDRT is invalid during the time.

0: When the SDA pin state disagrees with the bus interface outputs at the rise of SCL, or v SCL pin is driven low by another device.

1: When the SDA pin state disagrees with the bus interface outputs at the rise of SCL, or v SDA line is driven low by another device in i after the start condition instruction was execu

1	FNC1	0	R/W	Function Bit
0	FNC0	0	R/W	Cancels some restrictions on usage. For details section 16.6, Usage Notes.
				00: Restrictions on operation remaining in effect
				01: Setting prohibited
				10: Setting prohibited
				11: Restrictions on operation canceled

IIC_0 only is capable of formatless operation, as shown in figure 16.4.

The serial format is a non-addressing format with no acknowledge bit. This is shown in 16.5.

Figure 16.6 shows the I²C bus timing.

The symbols used in figures 16.3 to 16.6 are explained in table 16.6.

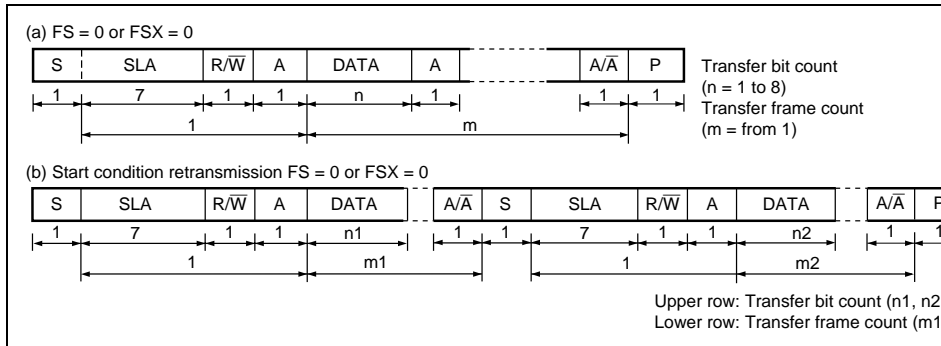


Figure 16.3 I²C Bus Data Format (I²C Bus Format)

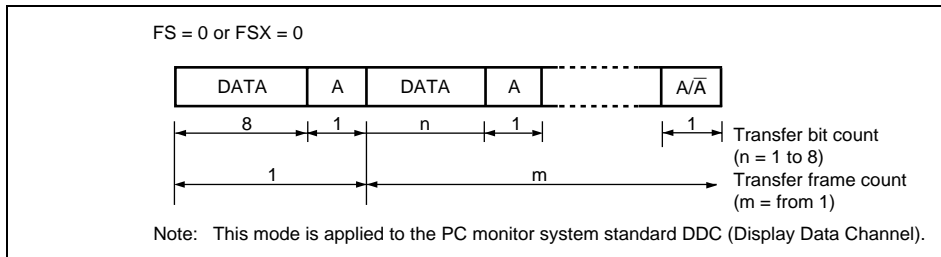


Figure 16.4 I²C Bus Data Format (Formatless) (IIC_0 Only)

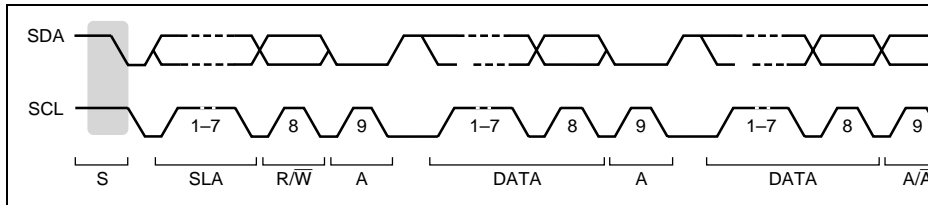


Figure 16.6 I²C Bus Timing

Table 16.6 I²C Bus Data Format Symbols

Legend

S	Start condition. The master device drives SDA from high to low while SCL is high.
SLA	Slave address. The master device selects the slave device.
R/W	Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.
A	Acknowledge. The receiving device drives SDA low to acknowledge a transmitted data byte. The slave device returns acknowledge in master transmit mode, and the master device returns acknowledge in master receive mode.)
DATA	Transferred data. The bit length of transferred data is set with the BC2 bit in ICMR. The MSB first or LSB first is switched with the MLS bit in ICMR.
P	Stop condition. The master device drives SDA from low to high while SCL is high.

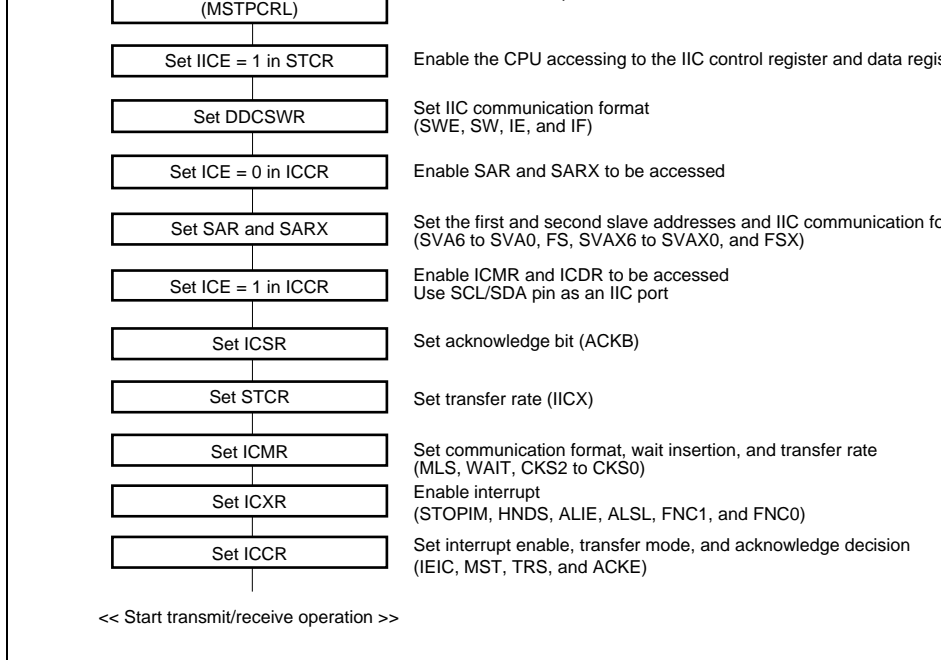


Figure 16.7 Sample Flowchart for IIC Initialization

Note: Be sure to modify the ICMR register after transmit/receive operation has been completed. If the ICMR register is modified during transmit/receive operation, bit counter BC0 will be modified erroneously, thus causing incorrect operation.

16.4.3 Master Transmit Operation

In I²C bus format master transmit mode, the master device outputs the transmit clock and data, and the slave device returns an acknowledge signal.

Figure 16.8 shows the sample flowchart for the operations in master transmit mode.

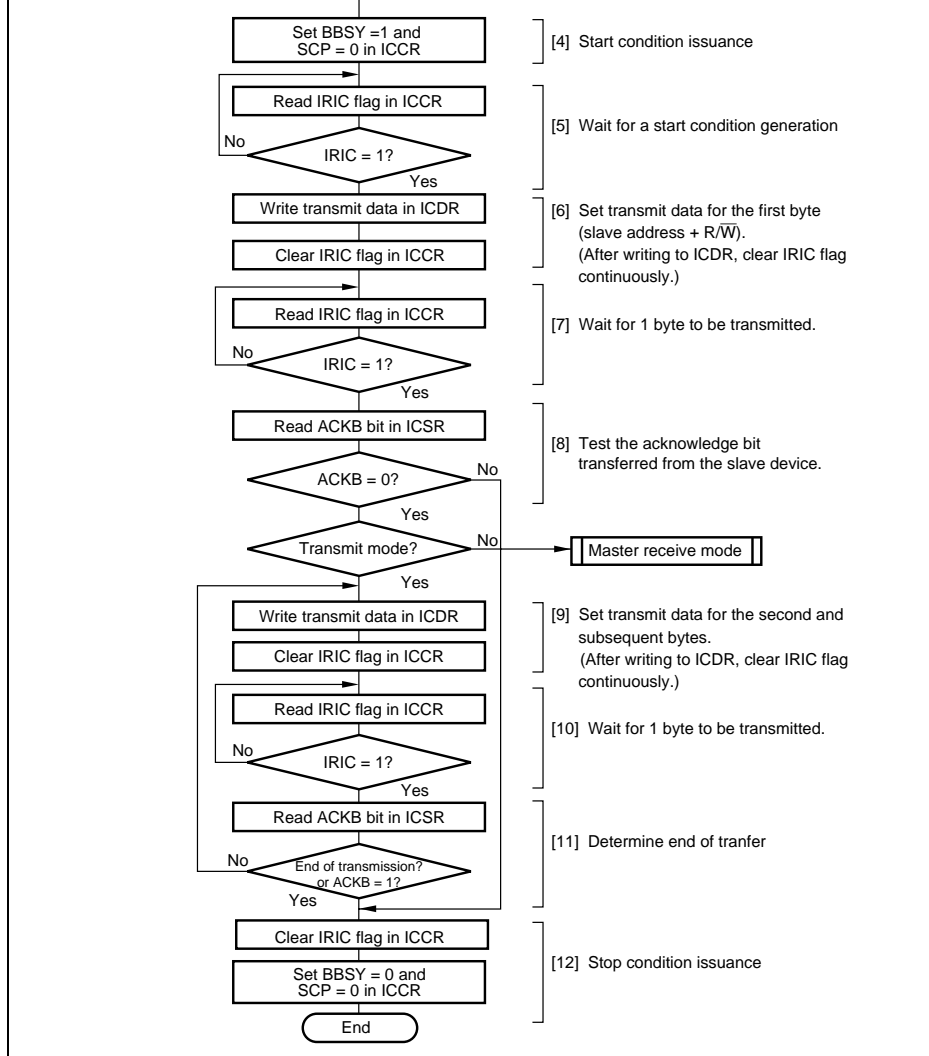


Figure 16.8 Sample Flowchart for Operations in Master Transmit Mode

5. Then the IRIC and IRTR flags are set to 1. If the IEIC bit in ICCR has been set to 1, an interrupt request is sent to the CPU.
6. Write the data (slave address + R/\overline{W}) to ICDR.

With the I²C bus format (when the FS bit in SAR or the FSX bit in SARX is 0), the data following the start condition indicates the 7-bit slave address and transmit/receive direction (R/\overline{W}).

To determine the end of the transfer, the IRIC flag is cleared to 0. After writing to ICDR, the IRIC flag is cleared continuously so no other interrupt handling routine is executed. If the time for transmission of one frame of data has passed before the IRIC clearing, the end of transmission cannot be determined. The master device sequentially sends the transmission clock and data written to ICDR. The selected slave device (i.e. the slave device with the matched slave address) drives SDA low at the 9th transmit clock pulse and returns an acknowledgment.
7. When one frame of data has been transmitted, the IRIC flag is set to 1 at the rise of the transmit clock pulse. After one frame has been transmitted, SCL is automatically fixed in synchronization with the internal clock until the next transmit data is written.
8. Read the ACKB bit in ICSR to confirm that ACKB is cleared to 0. When the slave device has not acknowledged (ACKB bit is 1), operate step [12] to end transmission, and retry the transmit operation.
9. Write the transmit data to ICDR.

As indicating the end of the transfer, the IRIC flag is cleared to 0. Perform the ICDR write and the IRIC flag clearing sequentially, just as in step [6]. Transmission of the next frame of data is performed in synchronization with the internal clock.
10. When one frame of data has been transmitted, the IRIC flag is set to 1 at the rise of the transmit clock pulse. After one frame has been transmitted, SCL is automatically fixed in synchronization with the internal clock until the next transmit data is written.
11. Read the ACKB bit in ICSR.

Confirm that the slave device has been acknowledged (ACKB bit is 0). When there is data to be transmitted, go to step [9] to continue the next transmission operation. When the slave device has not acknowledged (ACKB bit is set to 1), operate step [12] to end transmission.

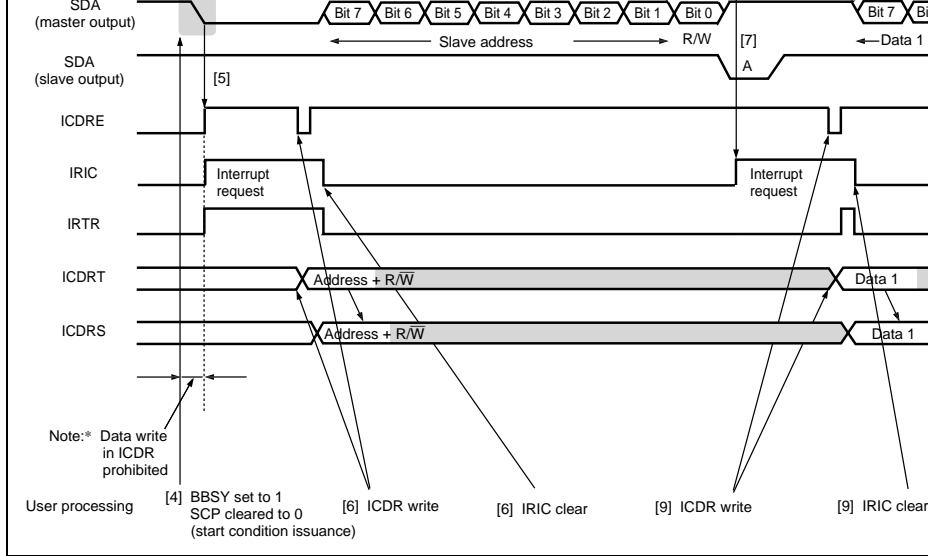


Figure 16.9 Example of Operation Timing in Master Transmit Mode (MLS = 1)

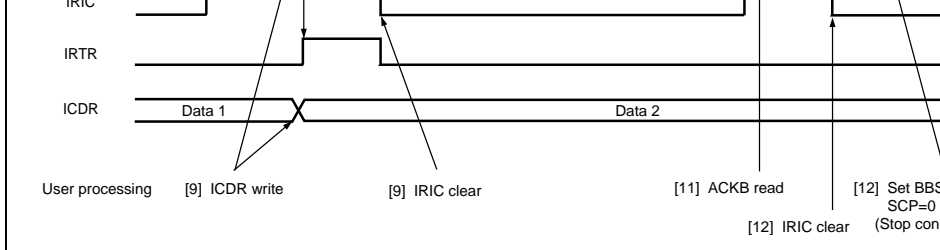


Figure 16.10 Example of Stop Condition Issuance Operation Timing in Master Transmit Mode (MLS = WAIT = 0)

16.4.4 Master Receive Operation

In I²C bus format master receive mode, the master device outputs the receive clock, receives data, and returns an acknowledge signal. The slave device transmits data.

The master device transmits data containing the slave address and $\overline{R/W}$ (1: read) in the following the start condition issuance in master transmit mode, selects the slave device, switches the mode for receive operation.

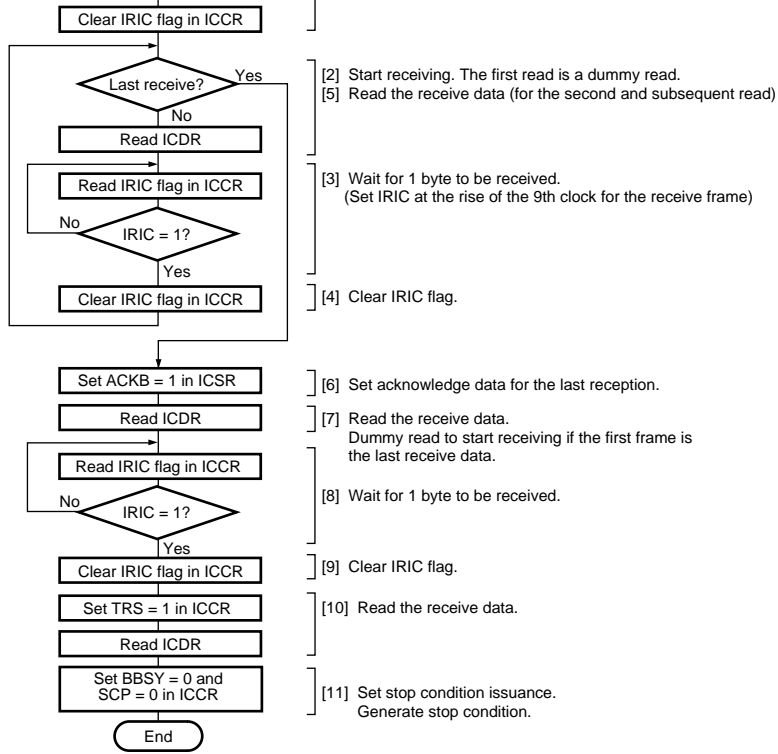


Figure 16.11 Sample Flowchart for Operations in Master Receive Mode (HNDS = 1)

2. When ICDR is read (dummy data read), reception is started, the receive clock is output in synchronization with the internal clock, and data is received. (Data from the SDA pin is sequentially transferred to ICDRS in synchronization with the rise of the receive clock pulse.)
3. The master device drives SDA low to return the acknowledge data at the 9th receive clock pulse. The receive data is transferred from ICDRS to ICDRR at the rise of the 9th receive clock pulse, setting the ICDRF, IRIC, and IRTR flags to 1. If the IEIC bit has been set to 1, an interrupt request is sent to the CPU.

The master device drives SCL low from the fall of the 9th receive clock pulse to the end of the reading.

4. Clear the IRIC flag to clear the wait state.
Go to step [6] to halt reception operation if the next frame is the last receive data.
5. Read ICDR receive data. This clears the ICDRF flag to 0. The master device outputs the receive clock continuously to receive the next data.

Data can be received continuously by repeating steps [3] to [5].

6. Set the ACKB bit to 1 so as to return the acknowledge data for the last reception.
7. Read ICDR receive data. This clears the ICDRF flag to 0. The master device outputs the receive clock to receive data.
8. When one frame of data has been received, the ICDRF, IRIC, and IRTR flags are set to 1 at the rise of the 9th receive clock pulse.
9. Clear the IRIC flag to 0.
10. Read ICDR receive data after setting the TRS bit. This clears the ICDRF flag to 0.
11. Clear the BBSY bit and SCP bit to 0 in ICCR. This changes SDA from low to high impedance, and generates the stop condition.

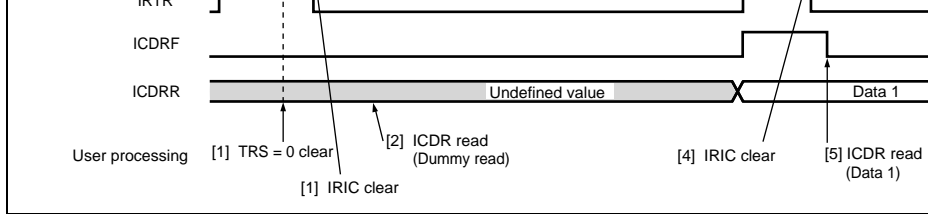


Figure 16.12 Example of Operation Timing in Master Receive Mode (ML = WAIT = 0, HNDS = 1)

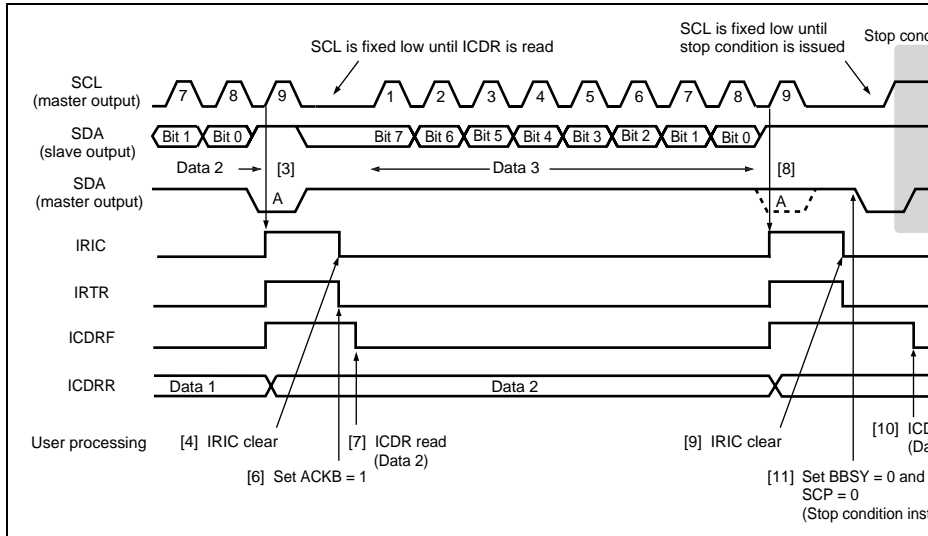


Figure 16.13 Example of Stop Condition Issuance Operation Timing in Master Receive Mode (ML = WAIT = 0, HNDS = 1)

Receive Operation Using the Wait Function:

Figures 16.14 and 16.15 show the sample flowcharts for the operations in master receive mode (WAIT = 1).

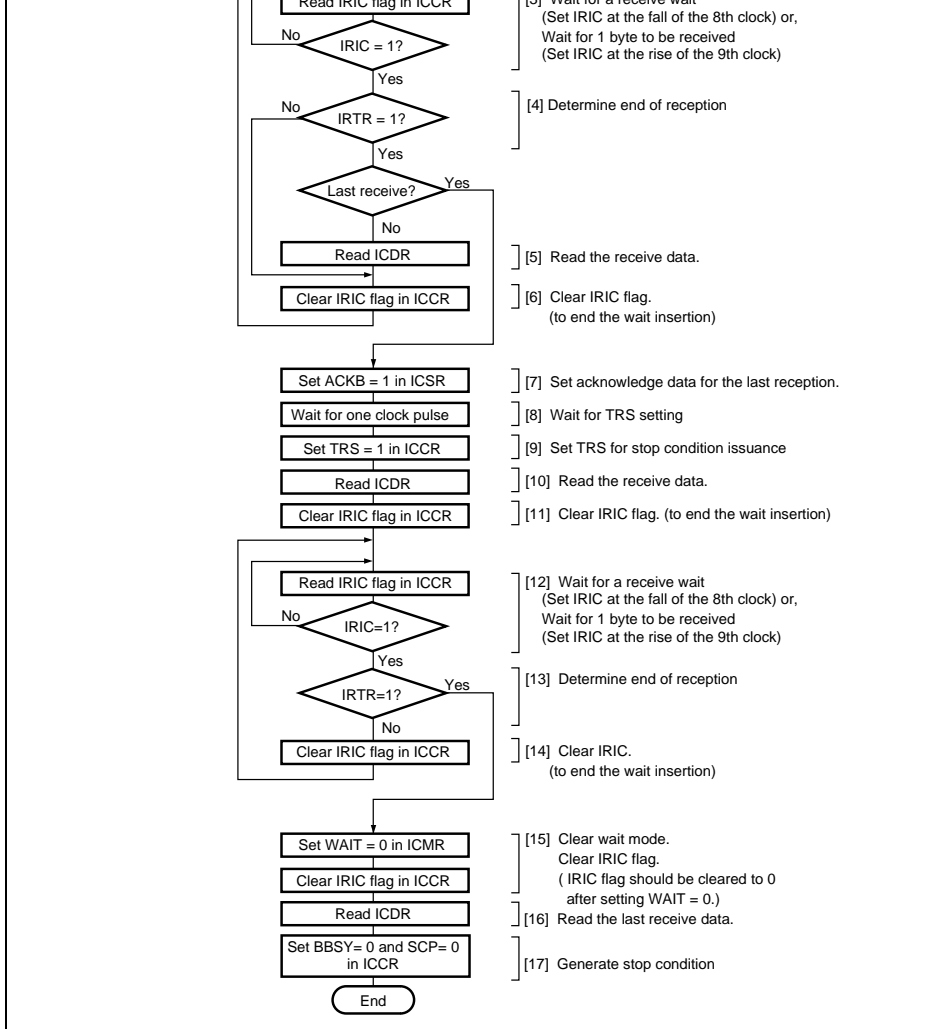


Figure 16.14 Sample Flowchart for Operations in Master Receive Mode (Receiving Multiple Bytes) (WAIT = 1)

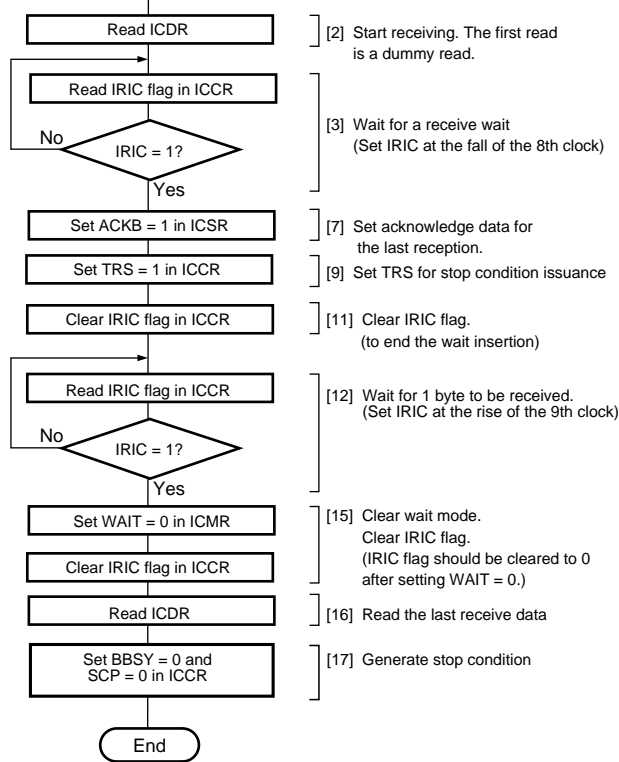


Figure 16.15 Sample Flowchart for Operations in Master Receive Mode (Receiving a Single Byte) (WAIT = 1)

- Clear the ACKB bit in ICSR to 0 to set the acknowledge data.
 - Clear the HNDS bit in ICXR to 0 to cancel the handshake function.
 - Clear the IRIC flag to 0, and then set the WAIT bit in ICMR to 1.
2. When ICDR is read (dummy data is read), reception is started, the receive clock is synchronized with the internal clock, and data is received.
 3. The IRIC flag is set to 1 in either of the following cases. If the IEIC bit in ICCR has 1, an interrupt request is sent to the CPU.
 - At the fall of the 8th receive clock pulse for one frame
SCL is automatically fixed low in synchronization with the internal clock until the IRIC flag clearing.
 - At the rise of the 9th receive clock pulse for one frame
The IRTR and ICDRF flags are set to 1, indicating that one frame of data has been received. The master device outputs the receive clock continuously to receive the data.
 4. Read the IRTR flag in ICSR.
 - If the IRTR flag is 0, execute step [6] to clear the IRIC flag to 0 to release the wait status.
 - If the IRTR flag is 1 and the next data is the last receive data, execute step [7] to handle the last receive data.
 5. If IRTR flag is 1, read ICDR receive data.
 6. Clear the IRIC flag. When the flag is set as the first case in step [3], the master device outputs the 9th clock and drives SDA low at the 9th receive clock pulse to return an acknowledge signal.

Data can be received continuously by repeating steps [3] to [6].

7. Set the ACKB bit in ICSR to 1 so as to return the acknowledge data for the last receive data.
8. After the IRIC flag is set to 1, wait for at least one clock pulse until the rise of the first receive clock pulse for the next receive data.
9. Set the TRS bit in ICCR to 1 to switch from receive mode to transmit mode. The TRS bit becomes valid when the rising edge of the next 9th clock pulse is input.
10. Read the ICDR receive data.

- RECEIVED. The master device outputs the receive clock continuously to receive data.
13. Read the IRTR flag in ICSR.
 - If the IRTR flag is 0, execute step [14] to clear the IRIC flag to 0 to release the wait state.
 - If the IRTR flag is 1 and data reception is complete, execute step [15] to issue the stop condition.
 14. If IRTR flag is 0, clear the IRIC flag to 0 to release the wait state.
 - Execute step [12] to read the IRIC flag to detect the end of reception.
 15. Clear the WAIT bit in CMR to cancel the wait mode.
 - Then, clear the IRIC flag. Clearing of the IRIC flag should be done while WAIT = 1. (If the WAIT bit is cleared to 0 after clearing the IRIC flag and then an instruction to issue the stop condition is executed, the stop condition may not be issued correctly.)
 16. Read the last ICDR receive data.
 17. Clear the BBSY bit and SCP bit to 0 in ICCR. This changes SDA from low to high. (When SDA is high, and generates the stop condition.

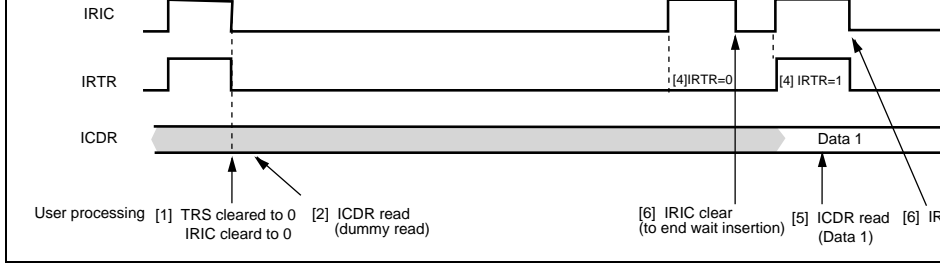


Figure 16.16 Example of Master Receive Mode Operation Timing
 (MLS = ACKB = 0, WAIT = 1)

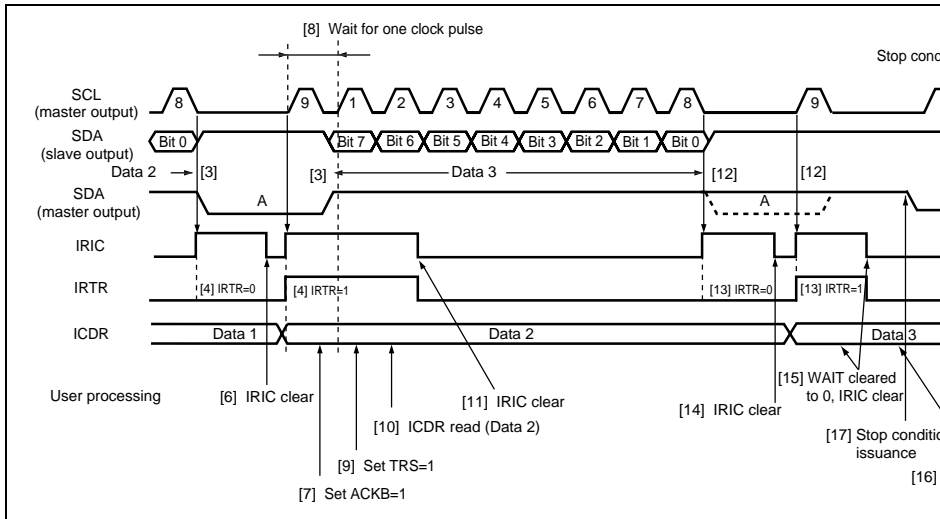


Figure 16.17 Example of Stop Condition Issuance Timing in Master Receive Mode
 (MLS = ACKB = 0, WAIT = 1)

Receive Operation Using the HNDS Function (HNDS = 1):

Figure 16.18 shows the sample flowchart for the operations in slave receive mode (HN

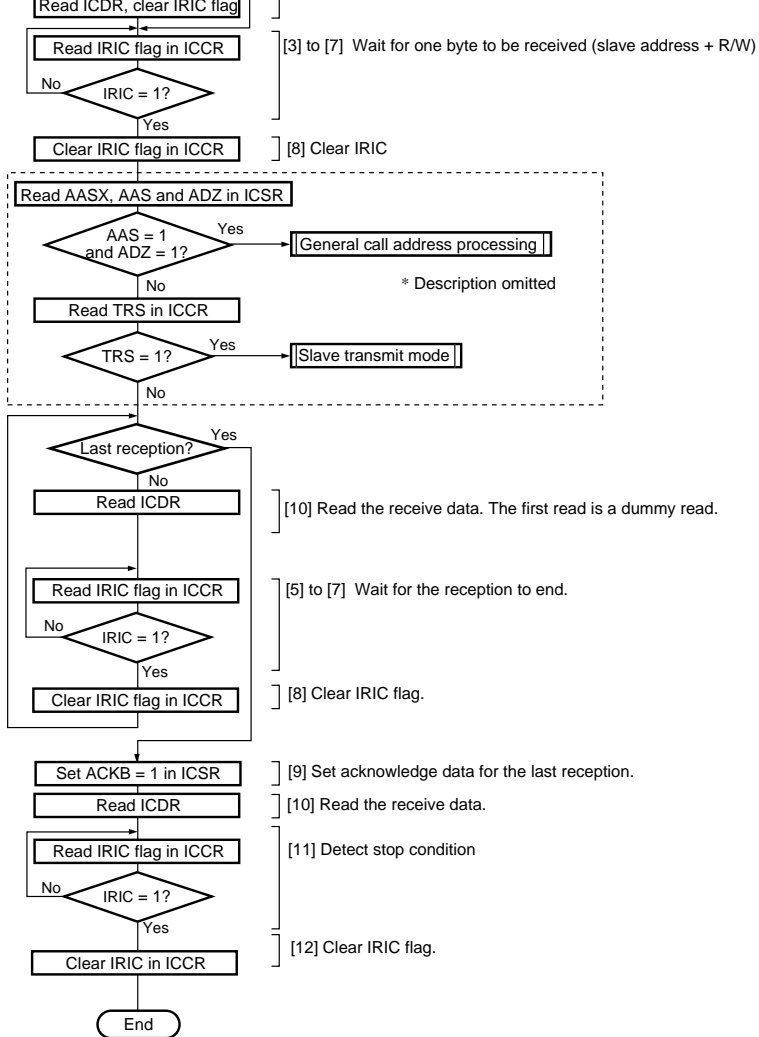


Figure 16.18 Sample Flowchart for Operations in Slave Receive Mode (HND)

- the IRIC flag to 0.
- When the start condition output by the master device is detected, the BBSY flag in the ICDR is set to 1. The master device then outputs the 7-bit slave address and transmit/receive data (R/W), in synchronization with the transmit clock pulses.
 - When the slave address matches in the first frame following the start condition, the slave device operates as the slave device specified by the master device. If the 8th data bit ($\overline{R/W}$) is 0, the TRS bit remains cleared to 0, and slave receive operation is performed. If the 8th data bit ($\overline{R/W}$) is 1, the TRS bit is set to 1, and slave transmit operation is performed. When the slave address does not match, receive operation is halted until the next start condition is detected.
 - At the 9th clock pulse of the receive frame, the slave device returns the data in the ICDR as an acknowledge signal.
 - At the rise of the 9th clock pulse, the IRIC flag is set to 1. If the IEIC bit has been set to 1, an interrupt request is sent to the CPU.
If the AASX bit has been set to 1, IRTR flag is also set to 1.
 - At the rise of the 9th clock pulse, the receive data is transferred from ICDRS to ICDR, setting the ICDRF flag to 1. The slave device drives SCL low from the fall of the 9th clock pulse until data is read from ICDR.
 - Confirm that the STOP bit is cleared to 0, and clear the IRIC flag to 0.
 - If the next frame is the last receive frame, set the ACKB bit to 1.
 - If ICDR is read, the ICDRF flag is cleared to 0, releasing the SCL bus line. This enables the master device to transfer the next data.

Receive operations can be performed continuously by repeating steps [5] to [10].

- When the stop condition is detected (SDA is changed from low to high when SCL is high), the BBSY flag is cleared to 0 and the STOP bit is set to 1. If the STOPIM bit has been set to 1, an interrupt request is sent to the CPU. If the AASX bit has been set to 1, IRTR flag is also set to 1.
- Confirm that the STOP bit is set to 1, and clear the IRIC flag to 0.

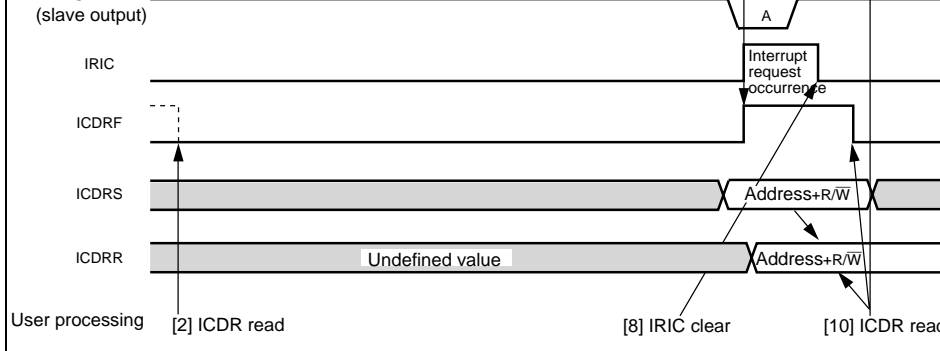


Figure 16.19 Example of Slave Receive Mode Operation Timing (1)
 (MLS = 0, HNDS= 1)

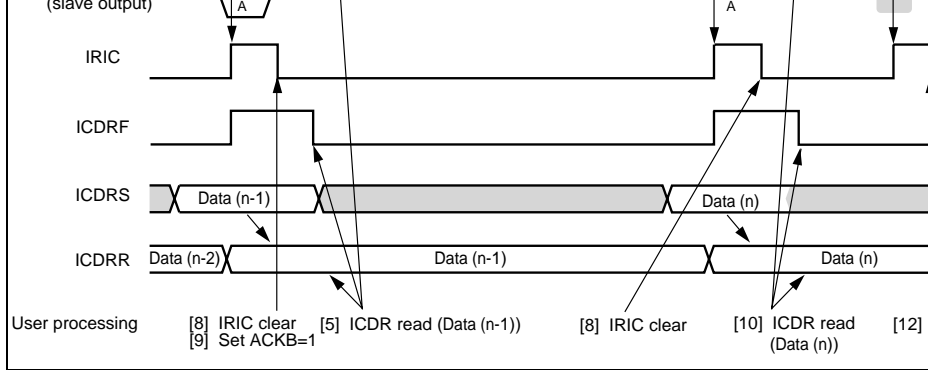


Figure 16.20 Example of Slave Receive Mode Operation Timing (2)
(MLS = 0, HNDS= 1)

Continuous Receive Operation:

Figure 16.21 shows the sample flowchart for the operations in slave receive mode (HNDS=1).

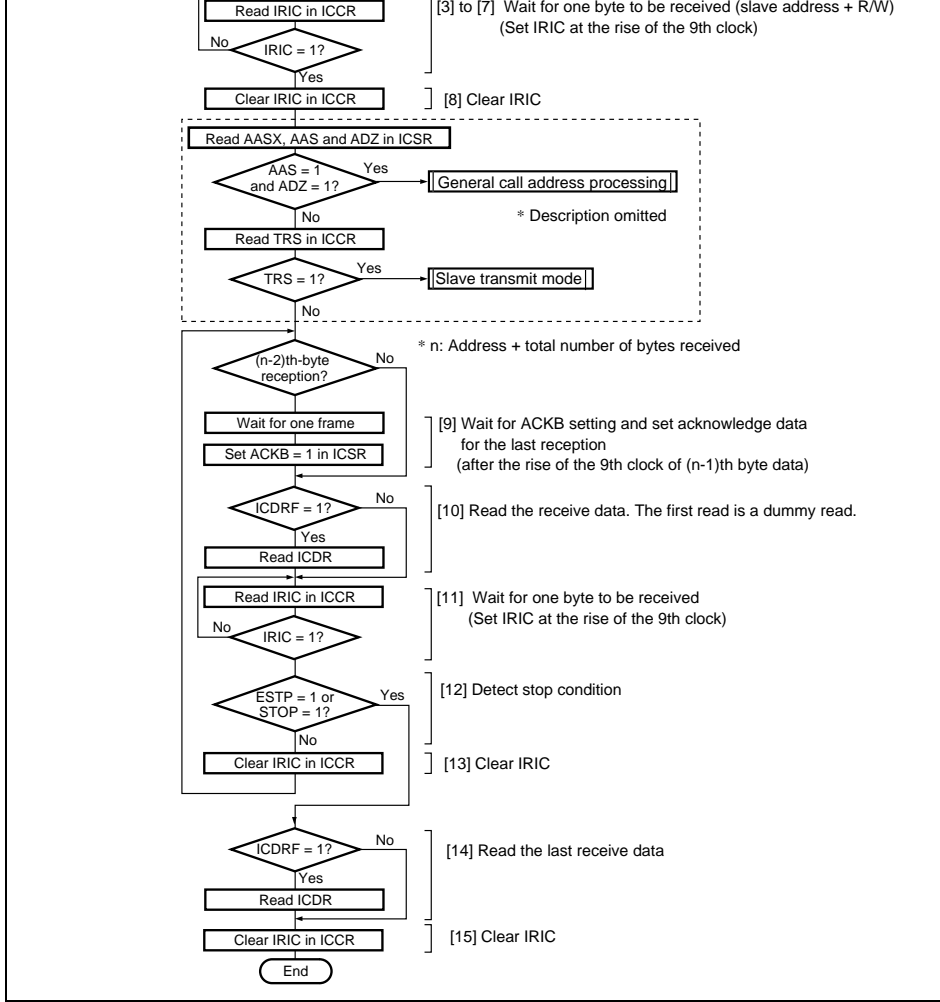


Figure 16.21 Sample Flowchart for Operations in Slave Receive Mode (HND)

- to 1. The master device then outputs the 7-bit slave address and transmit/receive direction (R/W) in synchronization with the transmit clock pulses.
4. When the slave address matches in the first frame following the start condition, the device operates as the slave device specified by the master device. If the 8th data bit (R/W) TRS bit remains cleared to 0, and slave transmit operation is performed. When the address does not match, receive operation is halted until the next start condition is detected.
 5. At the 9th clock pulse of the receive frame, the slave device returns the data in the receive data register as an acknowledge signal.
 6. At the rise of the 9th clock pulse, the IRIC flag is set to 1. If the IEIC bit has been set to 1, an interrupt request is sent to the CPU.
If the AASX bit has been set to 1, the IRTR flag is also set to 1.
 7. At the rise of the 9th clock pulse, the receive data is transferred from ICDRS to ICDR, setting the ICDRF flag to 1.
 8. Confirm that the STOP bit is cleared to 0 and clear the ICIC flag to 0.
 9. If the next read data is the third last receive frame, wait for at least one frame time before setting the ACKB bit. Set the ACKB bit after the rise of the 9th clock pulse of the second last receive frame.
 10. Confirm that the ICDRF flag is set to 1 and read ICDR. This clears the ICDRF flag.
 11. At the rise of the 9th clock pulse or when the receive data is transferred from IRDR to ICDR due to ICDR read operation, the IRIC and ICDRF flags are set to 1.
 12. When the stop condition is detected (SDA is changed from low to high when SCL is high), the BBSY flag is cleared to 0 and the STOP or ESTP flag is set to 1. If the STOPIM bit has been set to 1, the ICDR is cleared to 0, the IRIC flag is set to 1. In this case, execute step [14] to read the last receive data.
 13. Clear the IRIC flag to 0.

Receive operations can be performed continuously by repeating steps [9] to [13].

14. Confirm that the ICDRF flag is set to 1, and read ICDR.
15. Clear the IRIC flag.

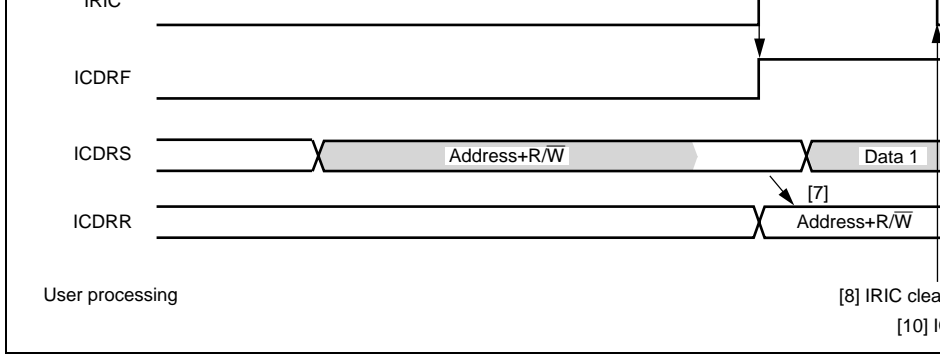


Figure 16.22 Example of Slave Receive Mode Operation Timing (1)
 (MLS = ACKB = 0, HNDS = 0)

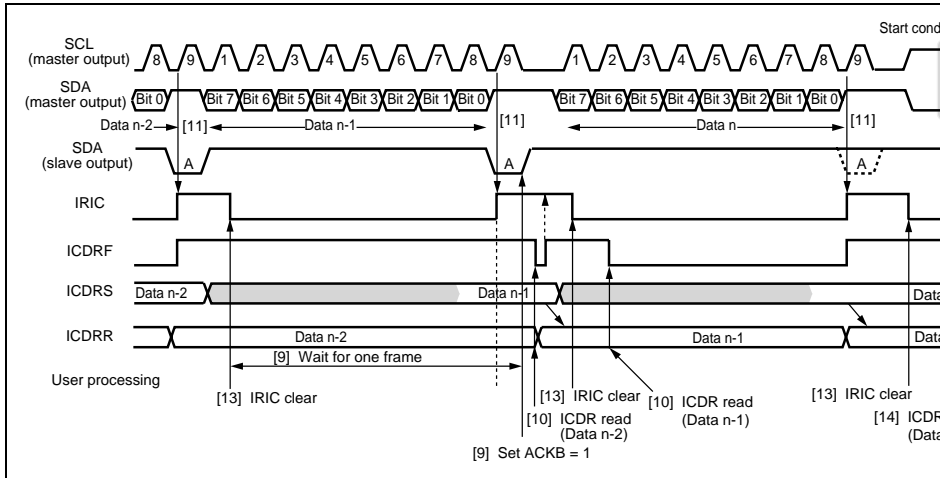


Figure 16.23 Example of Slave Receive Mode Operation Timing (2)
 (MLS = ACKB = 0, HNDS = 0)

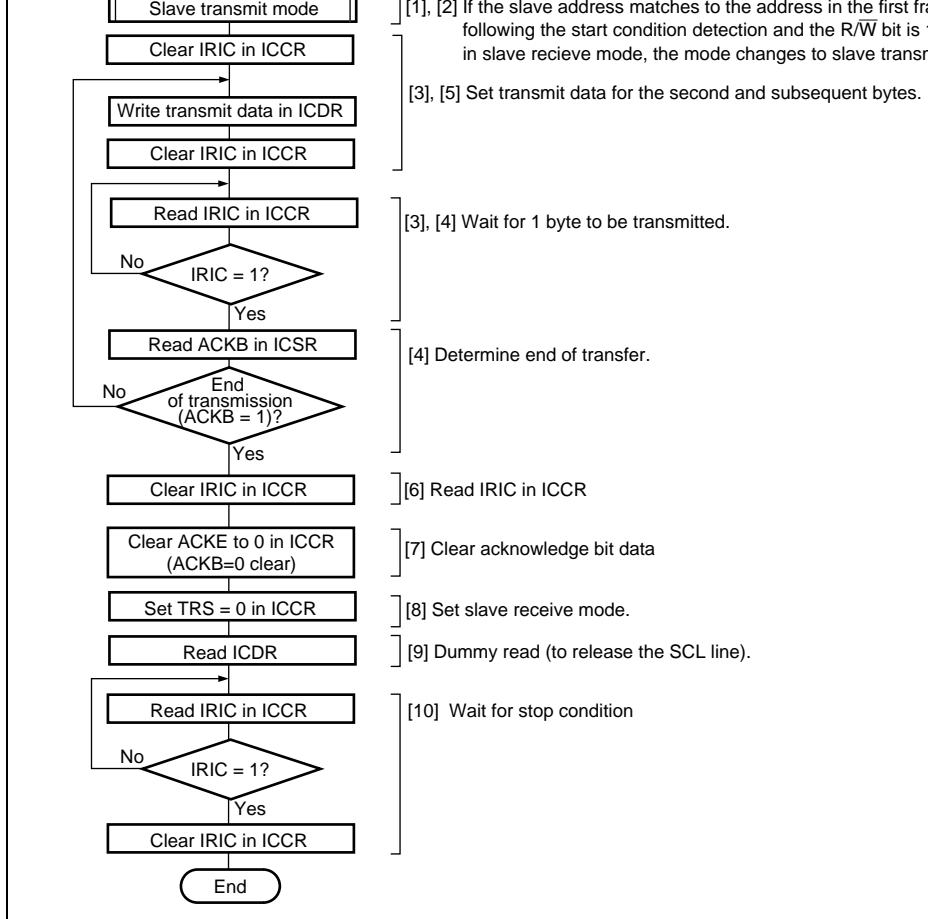


Figure 16.24 Sample Flowchart for Slave Transmit Mode

- transmit mode automatically. The IRIC flag is set to 1 at the rise of the 9th clock. The ICDRE flag is set to 1. The slave device drives SCL low from the fall of the transmit clock until ICDR data is written, to disable the master device to output the next transfer clock.
3. After clearing the IRIC flag to 0, write data to ICDR. At this time, the ICDRE flag is set to 1. The written data is transferred to ICDRS, and the ICDRE and IRIC flags are set to 0. The slave device sequentially sends the data written into ICDRS in accordance with the output by the master device.
The IRIC flag is cleared to 0 to detect the end of transmission. Processing from the ICDR register writing to the IRIC flag clearing should be performed continuously. Preventing interrupt processing from being inserted.
 4. The master device drives SDA low at the 9th clock pulse, and returns an acknowledge signal. As this acknowledge signal is stored in the ACKB bit in ICSR, this bit can be used to determine whether the transfer operation was performed successfully. When one frame has been transmitted, the IRIC flag in ICCR is set to 1 at the rise of the 9th transmit clock pulse. When the ICDRE flag is 0, the data written into ICDR is transferred to ICDRS. Transmission starts, and the ICDRE and IRIC flags are set to 1 again. If the ICDRE flag has been set to 1, this slave device drives SCL low from the fall of the transmit clock until the data is written to ICDR.
 5. To continue transmission, write the next data to be transmitted into ICDR. The ICDRE flag is cleared to 0. The IRIC flag is cleared to 0 to detect the end of transmission. Processing from the ICDR register writing to the IRIC flag clearing should be performed continuously. Preventing any other interrupt processing from being inserted.

Transmit operations can be performed continuously by repeating steps [4] and [5].

6. Clear the IRIC flag to 0.
7. To end transmission, clear the ACKE bit in ICCR to 0, to clear the acknowledge bit in ICSR, and clear the ACKB bit to 0.
8. Clear the TRS bit to 0 for the next address reception, to set slave receive mode.
9. Dummy-read ICDR to release SDA on the slave side.

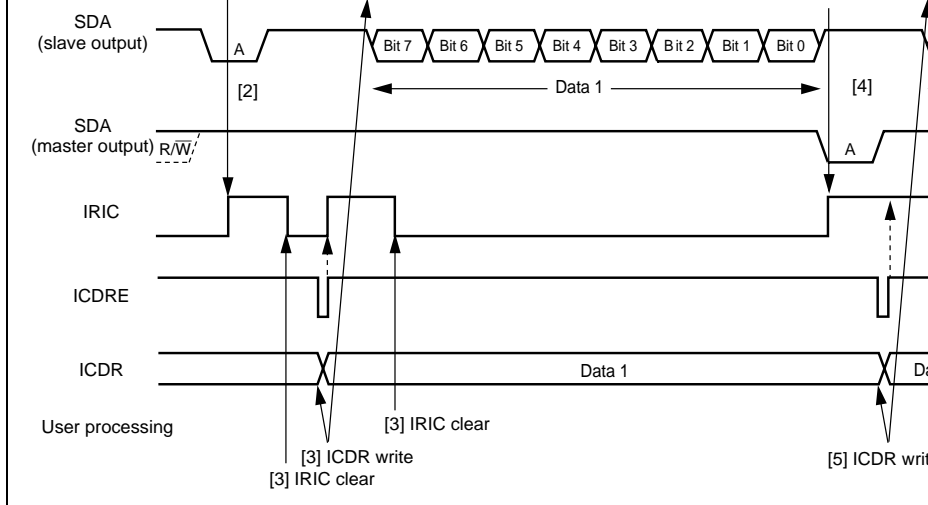
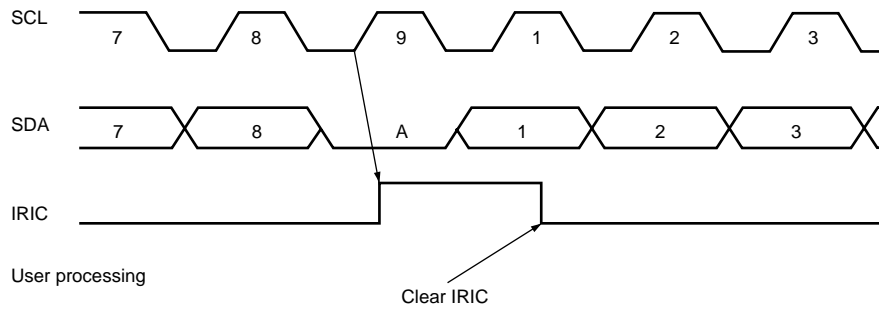
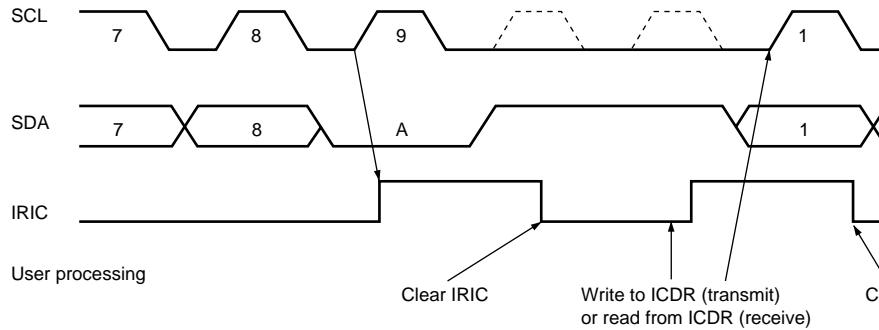


Figure 16.25 Example of Slave Transmit Mode Operation Timing (MLS = 0)



(a) Data transfer ends with ICDRE = 0 at transmission, or ICDRF = 0 at reception



(b) Data transfer ends with ICDRE = 1 at transmission, or ICDRF = 1 at reception

Figure 16.26 IRIC Setting Timing and SCL Control (1)

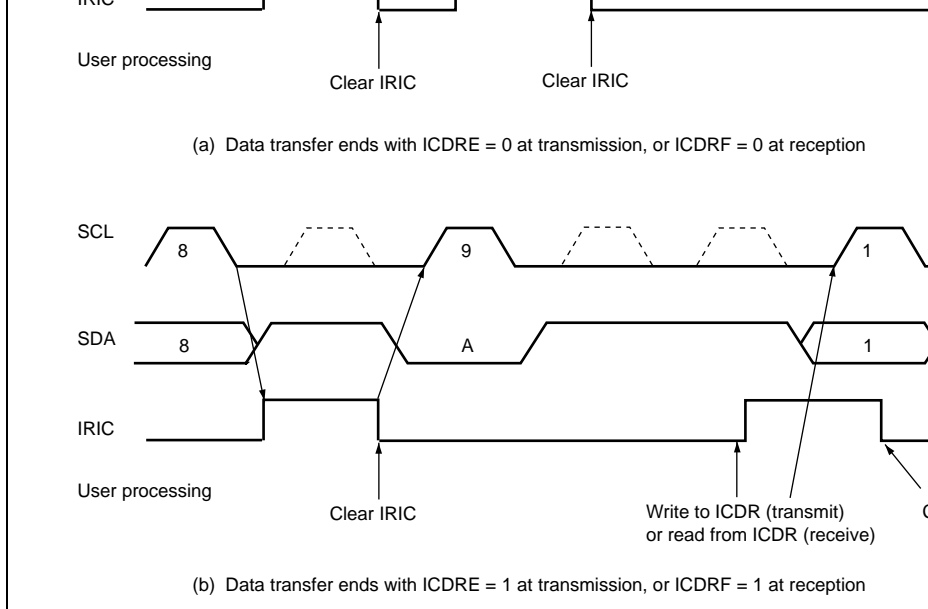


Figure 16.27 IRIC Setting Timing and SCL Control (2)

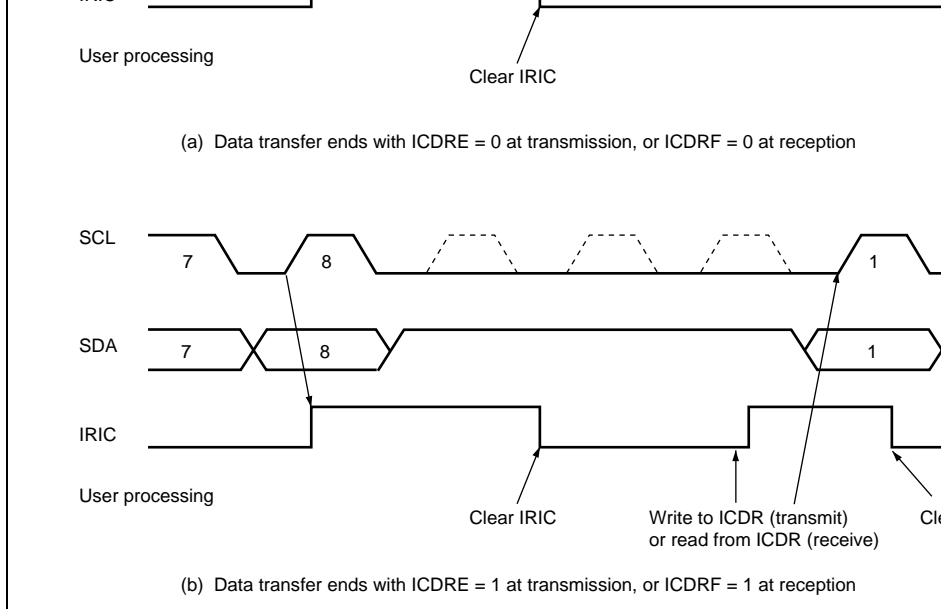


Figure 16.28 IRIC Setting Timing and SCL Control (3)

16.4.8 Automatic Switching from Formatless Mode to I²C Bus Format

Setting the SW bit to 1 in DDCSWR enables formatless mode to be selected as the IIC operating mode. Switching from formatless mode to the I²C bus format (slave mode) is automatically when a falling edge is detected on the SCL pin.

The following four preconditions are necessary for this operation:

- A common data pin (SDA) for formatless and I²C bus format operation
- Separate clock pins for formatless operation (VSYNCl) and I²C bus format operation
- A fixed 1 level for the SCL pin during formatless operation (the SCL pin does not change level)

bit to 1 or clear it to 0 according to the transfer direction (transmission or reception) in mode, then set the SW bit to 1. After automatic switching from formatless mode to the format (slave mode), the TRS bit is automatically cleared to 0 in order to wait for slave reception.

If a falling edge is detected on the SCL pin during formatless operation, the mode of the interface is immediately switched to I²C bus format before a stop condition is detected.

16.4.9 Operation Using DTC

This LSI provides the DTC to allow continuous data transfer. The DTC is initiated when the DTCEN bit is set to 1, which is one of the two interrupt flags (IRTR and IRIC). When the ACK bit is set to 1, the ICDRE, IRIC, and IRTR flags are set at the end of data transmission regardless of the acknowledge bit value. If the ACK bit is 0, the ICDRE, IRIC, and IRTR flags are set when data transmission is completed with the acknowledge bit value of 0, and if the ACK bit is 1, the IRIC flag is set when data transmission is completed with the acknowledge bit value of 1.

When initiated, the DTC transfers specified number of bytes, clears the ICDRE, IRIC, and IRTR flags to 0. Therefore, no interrupt is generated during continuous data transfer; however, when data transmission is completed with the acknowledge bit value of 1 when the ACK bit is 1, the DTC is not initiated, thus allowing an interrupt to be generated if enabled.

The acknowledge bit may indicate specific events such as completion of receive data for some receiving devices, and for other receiving devices, the acknowledge bit may be 0, indicating no specific events.

The I²C bus format provides for selection of the slave device and transfer direction by the slave address and the R/ \bar{W} bit, confirmation of reception with the acknowledge bit of the last frame, and so on. Therefore, continuous data transfer using the DTC must be done in conjunction with CPU processing by means of interrupts.

Table 16.7 shows some examples of processing using the DTC. These examples assume that the number of transfer data bytes is known in slave mode.

		CPU (ICDR read)		
Actual data transmission/reception	Transmission by DTC (ICDR write)	Reception by DTC (ICDR read)	Transmission by DTC (ICDR write)	Reception by DTC (ICDR read)
Dummy data (H'FF) write	—	—	Processing by DTC (ICDR write)	—
Last frame processing	Not necessary	Reception by CPU (ICDR read)	Not necessary	Reception by CPU (ICDR read)
Transfer request processing after last frame processing	1st time: Clearing by CPU 2nd time: Stop condition issuance by CPU	Not necessary	Automatic clearing on detection of stop condition during transmission of dummy data (H'FF)	Not necessary
Setting of number of DTC transfer data frames	Transmission: Actual data count + 1 (+1 equivalent to slave address + R/W bits)	Reception: Actual data count	Transmission: Actual data count + 1 (+1 equivalent to dummy data (H'FF))	Reception: Actual data count

16.4.10 Noise Canceler

The logic levels at the SCL and SDA pins are routed through noise cancelers before being processed internally. Figure 16.29 shows a block diagram of the noise canceler.

The noise canceler consists of two cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the system clock, but is not passed forward to the next circuit until the outputs of both latches agree. If they do not agree, the previous value is held.

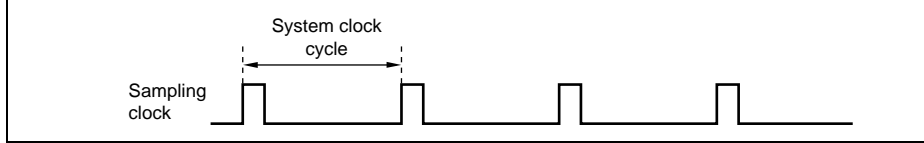


Figure 16.29 Block Diagram of Noise Canceler

16.4.11 Initialization of Internal State

The IIC has a function for forcible initialization of its internal state if a deadlock occurs during communication.

Initialization is executed in accordance with the setting of bits CLR3 to CLR0 in DDCCSR, clearing ICE bit. For details on the setting of bits CLR3 to CLR0, see section 16.3.7, IIC Register (DDCCSR).

Scope of Initialization: The initialization executed by this function covers the following:

- ICDRE and ICDRF internal flags
- Transmit/receive sequencer and internal operating clock counter
- Internal latches for retaining the output state of the SCL and SDA pins (wait, clock output, etc.)

The following items are not initialized:

- Actual register values (ICDR, SAR, SARX, ICMR, ICCR, ICSR, ICXR (except for ICDRE and ICDRF flags), DDCCSR)
- Internal latches used to retain register read information for setting/clearing flags in ICMR, ICSR, and DDCCSR
- The value of the ICMR bit counter (BC2 to BC0)
- Generated interrupt sources (interrupt sources transferred to the interrupt controller)

- using an MOV instruction. Do not use a bit manipulation instruction such as BCLR.
- Similarly, when clearing is required again, all the bits must be written to simultaneously in accordance with the setting.
 - If a flag clearing setting is made during transmission/reception, the IIC module will stop transmitting/receiving at that point and the SCL and SDA pins will be released. When transmission/reception is started again, register initialization, etc., must be carried out as necessary to enable correct communication as a system.

The value of the BBSY bit cannot be modified directly by this module clear function, but when a stop condition pin waveform is generated according to the state and release timing of the SDA pins, the BBSY bit may be cleared as a result. Similarly, state switching of other IIC flags may also have an effect.

To prevent problems caused by these factors, the following procedure should be used when initializing the IIC state.

1. Execute initialization of the internal state according to the setting of bits CLR3 to CLR0 and ICE bit clearing.
2. Execute a stop condition issuance instruction (write 0 to BBSY and SCP) to clear the BBSY bit to 0, and wait for two transfer rate clock cycles.
3. Re-execute initialization of the internal state according to the setting of bits CLR3 to CLR0 and ICE bit clearing.
4. Initialize (re-set) the IIC registers.

Channel	Name	Enable Bit	Interrupt Source	Interrupt Flag	DTC Activation
0	IIC10	IEIC	I ² C bus interface interrupt request	IRIC	Possible
	DDCSWI	IE	Format automatic switch interrupt	IF	Not possible
1	IIC11	IEIC	I ² C bus interface interrupt request	IRIC	Possible

16.6 Usage Notes

- In master mode, if an instruction to generate a start condition is issued and then an instruction to generate a stop condition is issued before the start condition is output to the I²C bus, the stop condition will be output correctly. To output the start condition followed by the stop condition after issuing the instruction that generates the start condition, read DR in each I²C channel pin, and check that SCL and SDA are both low. The pin states can be monitored by reading DR even if the ICE bit is set to 1. Then issue the instruction that generates the stop condition. Note that SCL may not yet have gone low when BBSY is cleared to 0.
- Either of the following two conditions will start the next transfer. Pay attention to the conditions when accessing to ICDR.
 - Write to ICDR when ICE = 1 and TRS = 1 (including automatic transfer from ICDRS)
 - Read from ICDR when ICE = 1 and TRS = 0 (including automatic transfer from ICRR)
- Table 16.9 shows the timing of SCL and SDA outputs in synchronization with the clock. Timings on the bus are determined by the rise and fall times of signals affected by bus load capacitance, series resistance, and parallel resistance.

Retransmission start condition output setup time	t_{STASO}	$1t_{SCLO}$	ns
Stop condition output setup time	t_{STOSO}	$0.5t_{SCLO} + 2t_{cyc}$	ns
Data output setup time (master)	t_{SDASO}	$1t_{SCLLO} - 3t_{cyc}$	ns
Data output setup time (slave)		$1t_{SCLL} - (6t_{cyc} \text{ or } 12t_{cyc}^*)$	
Data output hold time	t_{SDAHO}	$3t_{cyc}$	ns

Note: * $6t_{cyc}$ when IICX is 0, $12t_{cyc}$ when 1.

- SCL and SDA inputs are sampled in synchronization with the internal clock. The A therefore depends on the system clock cycle t_{cyc} , as shown in section 28, Electrical Characteristics. Note that the I²C bus interface AC timing specifications will not be system clock frequency of less than 5 MHz.
- The I²C bus interface specification for the SCL rise time t_{sr} is 1000 ns or less (300 ns speed mode). In master mode, the I²C bus interface monitors the SCL line and synchronizes one bit at a time during communication. If t_{sr} (the time for SCL to go from low to V_{CC}) is longer than the time determined by the input clock of the I²C bus interface, the high period of SCL is extended. The SCL rise time is determined by the pull-up resistance and load capacitance on the SCL line. To insure proper operation at the set transfer rate, adjust the pull-up resistor and load capacitance so that the SCL rise time does not exceed the values given in t

17.5 t_{cyc}	Standard mode	1000	100	1000	1000	1000
	High-speed mode	300	300	300	300	300

6. The I²C bus interface specifications for the SCL and SDA rise and fall times are t_{rise} and 300 ns. The I²C bus interface SCL and SDA output timing is prescribed by t_{cyc} in table 16.9. However, because of the rise and fall times, the I²C bus interface specifications cannot be satisfied at the maximum transfer rate. Table 16.11 shows output timing calculations at different operating frequencies, including the worst-case influence of rise and fall times. t_{BUFO} fails to meet the I²C bus interface specifications at any frequency. The solution is to provide coding to secure the necessary interval (approximately 1 μ s) between a stop condition and issuance of a start condition, or (b) to select devices whose input timing permits this output timing for use as slave devices connected to the I²C bus.
- t_{SCLLO} in high-speed mode and t_{STASO} in standard mode fail to satisfy the I²C bus interface specifications for worst-case calculations of t_{sr}/t_{sr} . Possible solutions that should be considered include (a) adjusting the rise and fall times by means of a pull-up resistor and capacitor, (b) reducing the transfer rate to meet the specifications, or (c) selecting devices whose input timing permits this output timing for use as slave devices connected to the I²C bus.



t_{SCLLO}	$0.5 t_{SCLLO} (-t_{Sr})$	Standard mode	-250	4700	4750	4750	4750	4750
		High-speed mode	-250	1300	1000 ^{*1}	1000 ^{*1}	1000 ^{*1}	1000
t_{BUFO}	$0.5 t_{SCLLO} - 1 t_{cyc} (-t_{Sr})$	Standard mode	-1000	4700	3800 ^{*1}	3875 ^{*1}	3900 ^{*1}	3938
		High-speed mode	-300	1300	750 ^{*1}	825 ^{*1}	850 ^{*1}	888 ^{*1}
t_{STAHO}	$0.5 t_{SCLLO} - 1 t_{cyc} (-t_{Sr})$	Standard mode	-250	4000	4550	4625	4650	4688
		High-speed mode	-250	600	800	875	900	938
t_{STASO}	$1 t_{SCLLO} (-t_{Sr})$	Standard mode	-1000	4700	9000	9000	9000	9000
		High-speed mode	-300	600	2200	2200	2200	2200
t_{STOSO}	$0.5 t_{SCLLO} + 2 t_{cyc} (-t_{Sr})$	Standard mode	-1000	4000	4400	4250	4200	4125
		High-speed mode	-300	600	1350	1200	1150	1075
t_{SDASO} (master)	$1 t_{SCLLO}^{*3} - 3 t_{cyc} (-t_{Sr})$	Standard mode	-1000	250	3100	3325	3400	3513
		High-speed mode	-300	100	400	625	700	813
t_{SDASO} (slave)	$1 t_{SCLL}^{*3} - 12 t_{cyc}^{*2} (-t_{Sr})$	Standard mode	-1000	250	1300	2200	2500	2950
		High-speed mode	-300	100	-1400 ^{*1}	-500 ^{*1}	-200 ^{*1}	250
t_{SDAHO}	$3 t_{cyc}$	Standard mode	0	0	600	375	300	188
		High-speed mode	0	0	600	375	300	188

Notes: 1. Does not meet the I²C bus interface specification. Remedial action such as the following is necessary: (a) secure a start/stop condition issuance interval; (b) adjust the rise and fall times by means of a pull-up resistor and capacitive load; (c) reduce the transfer rate; (d) select slave devices whose input timing permits this output timing.

The values in the above table will vary depending on the settings of the IICX bit (CKS0 to CKS2). Depending on the frequency it may not be possible to achieve the maximum transfer rate; therefore, whether or not the I²C bus interface specification is met must be determined in accordance with the actual setting conditions.

- Value when the IICX bit is set to 1. When the IICX bit is cleared to 0, the value is $(t_{SCLL} - 6t_{cyc})$.
- Calculated using the I²C bus specification values (standard mode: 4700 ns min., high-speed mode: 1300 ns min.).

BBSY bit in ICCR is cleared to 0, the stop condition has been generated, and the bus released, then read ICDR with TRS cleared to 0.

Note that if the receive data (ICDR data) is read in the interval between execution of the instruction for issuance of the stop condition (writing of 0 to BBSY and SCP in ICCR) and the actual generation of the stop condition, the clock may not be output correctly in subsequent master transmission.

Clearing of the MST bit after completion of master transmission/reception, or other modification of IIC control bits to change the transmit/receive operating mode or settings, must be performed during interval (a) in figure 16.30 (after confirming that the BBSY bit in ICCR has been cleared to 0).

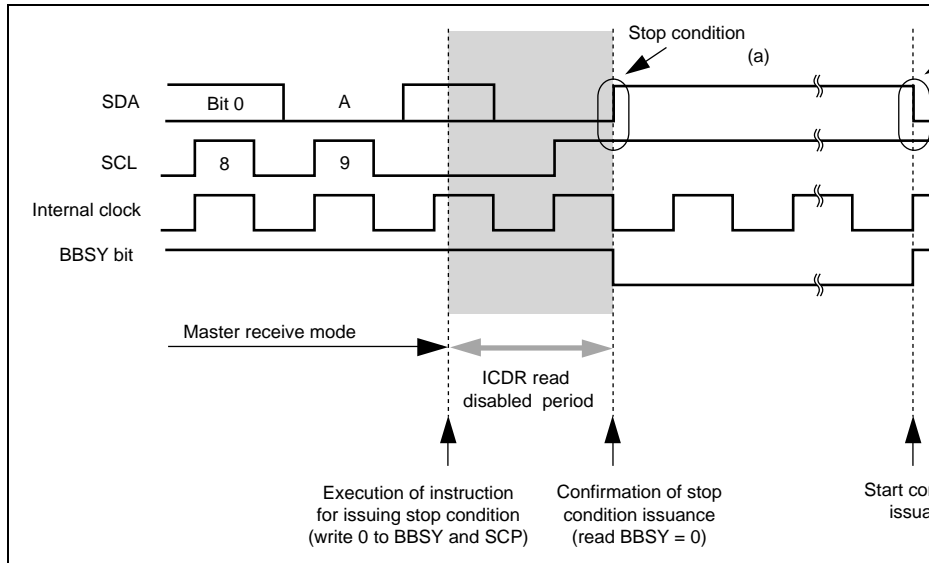


Figure 16.30 Notes on Reading Master Receive Data

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits in the IICXCR.

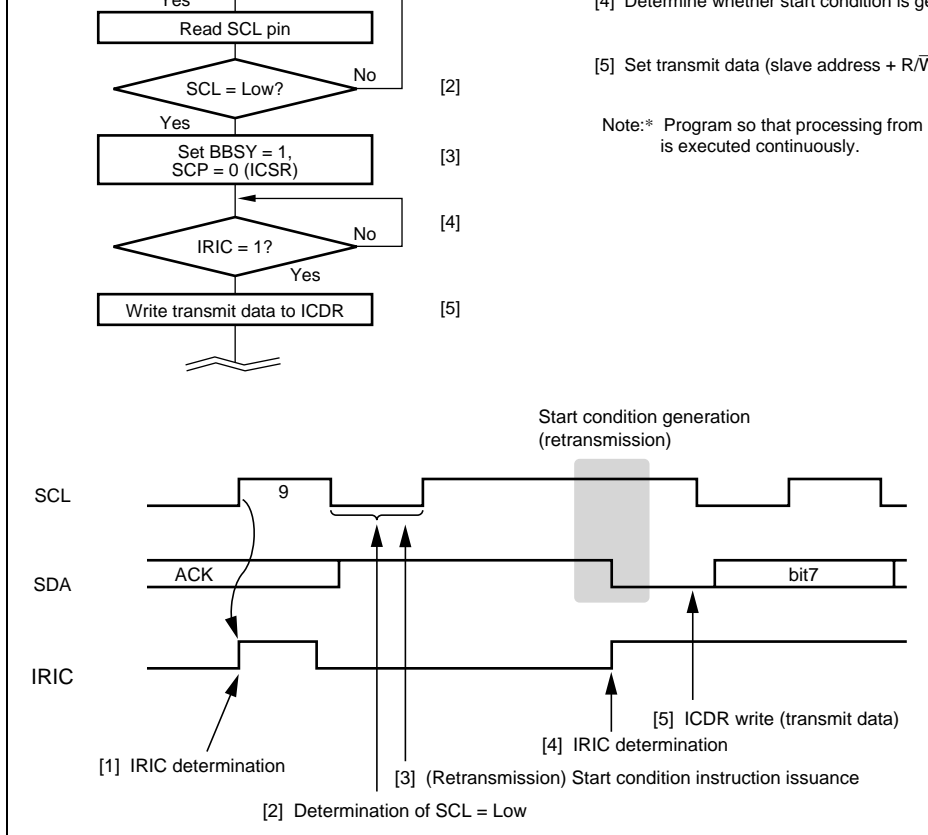


Figure 16.31 Flowchart for Start Condition Issuance Instruction for Retransmission Timing

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits in the ICSR.

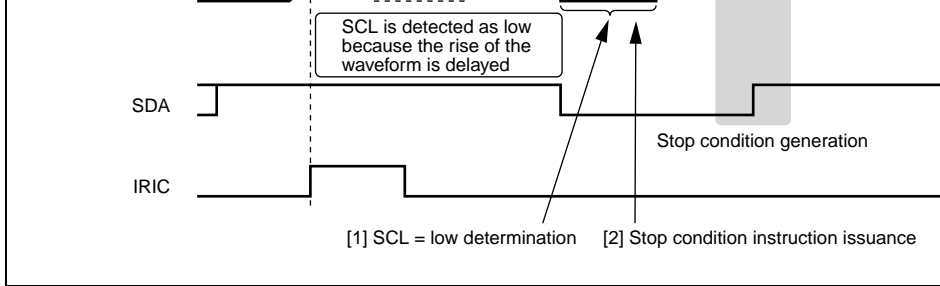


Figure 16.32 Stop Condition Issuance Timing

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits to 1 in the ICXR.

— Error phenomenon

Normally, WAIT State will be cancelled by clearing the IRIC flag bit from 1 to 0 on the fall of the 8th clock in WAIT State. In this case, if the IRIC flag bit is cleared before the 7th clock fall and the 8th clock fall, the IRIC flag clear-data will be retained in the WAIT State. Therefore, the WAIT State will be cancelled right after WAIT insertion on 8th clock.

— Restrictions

Please clear the IRIC flag before the rise of the 7th clock (the counter value of BC2 through BC0 should be 2 or greater), after the IRIC flag is set to 1 on the rise of the 8th clock.

If the IRIC flag-clear is delayed due to the interrupt or other processes and the counter is turned to 1 or 0, please confirm the SCL pins are in L' state after the value of BC2 through BC0 is turned to 0, and clear the IRIC flag. (See figure 16.33)

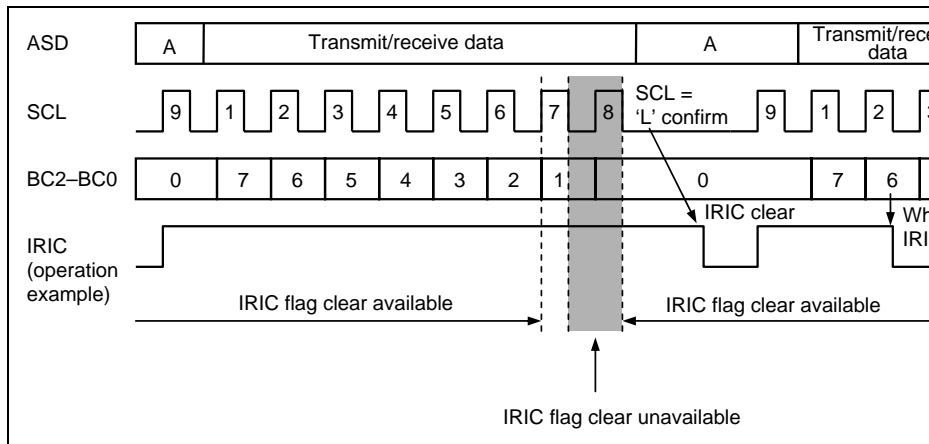


Figure 16.33 IRIC Flag Clear Timing on WAIT Operation

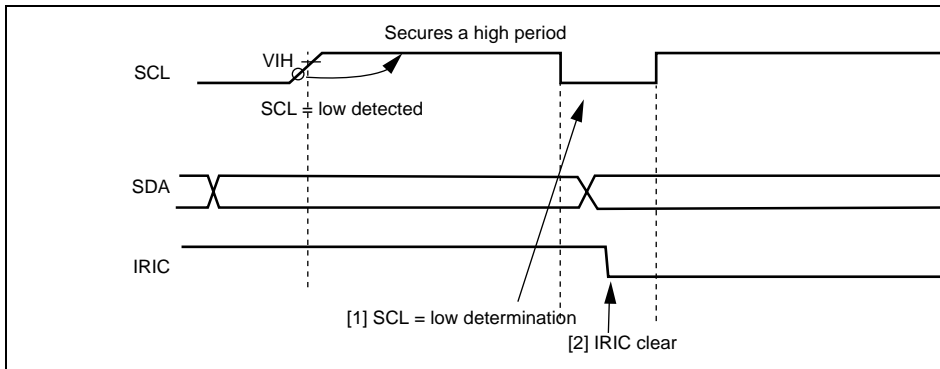


Figure 16.34 IRIC Flag Clearing Timing When WAIT = 1

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits to 1 in the ICXR.

the receive operation of the next slave address.

- Monitor the BC2 to BC0 bit counter in ICMR; when the count is 000 (8th or 9th pulse), wait for at least two transfer clock times in order to read ICDR or read/write ICCR during the time other than the shaded time.

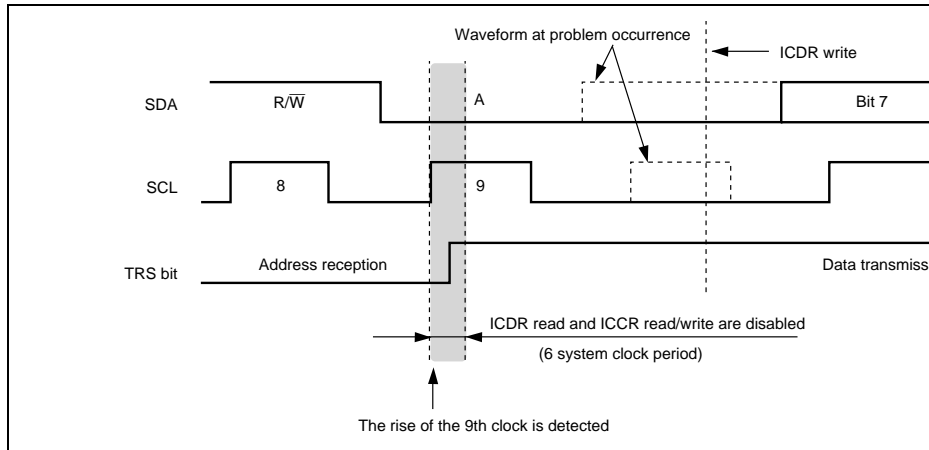


Figure 16.35 ICDR Read and ICCR Access Timing in Slave Transmitter Mode

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits in the ICDR register.

(transmit mode) internally and thus the acknowledge bit is not transmitted after the
 been received at the 9th clock pulse.

To receive the address in slave mode, clear the TRS bit to 0 during the time indicated in
 figure 16.36. To release the SCL low level that is held by means of the wait function
 mode, clear the TRS bit to and then dummy-read ICDR.

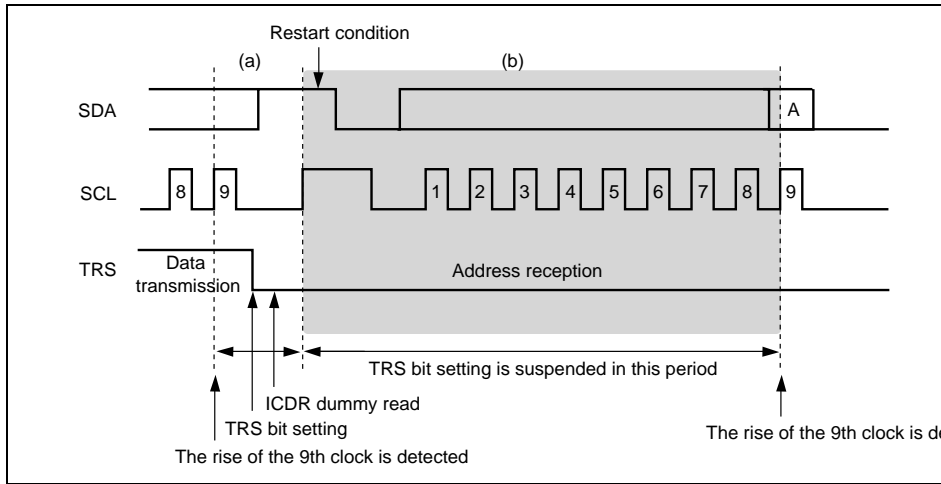


Figure 16.36 TRS Bit Set Timing in Slave Mode

Note: This restriction on usage can be canceled by setting the FNC1 and FNC0 bits to 1 in the ICXR.

figure 16.37.)

In multi-master mode, a bus conflict could happen. When The I²C bus interface is master mode, check the state of the AL bit in the ICSR register every time after one data has been transmitted or received.

When arbitration is lost during transmitting the second frame or subsequent frame, avoidance measures.

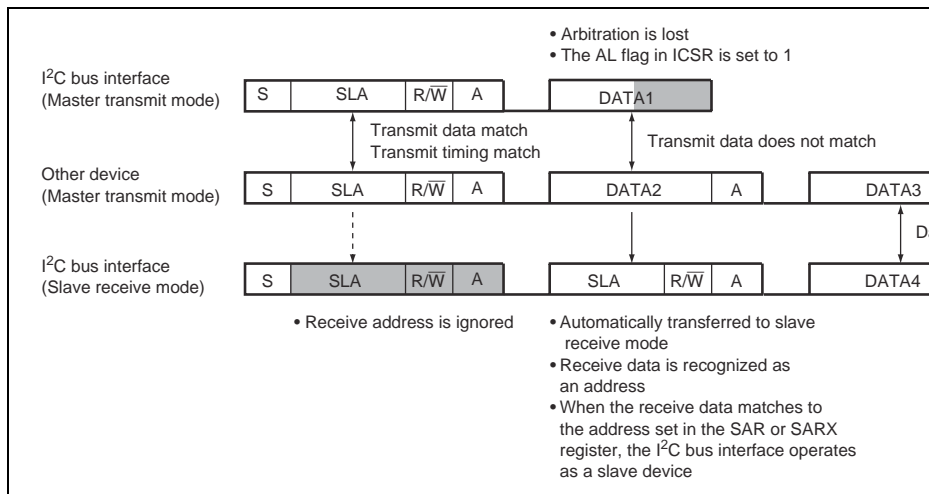


Figure 16.37 Diagram of Erroneous Operation when Arbitration is Lost

Though it is prohibited in the normal I²C protocol, the same problem may occur when the AL bit is erroneously set to 1 and a transition to master mode is occurred during data transmission or reception in slave mode. In multi-master mode, pay attention to the setting of the AL bit when a bus conflict may occur. In this case, the MST bit in the ICCR register should be set to 1 according to the order below.

- (a) Make sure that the BBSY flag in the ICCR register is 0 and the bus is free before setting the MST bit.
- (b) Set the MST bit to 1.

ICDR is accessed correctly. To access ICDR correctly, read ICDR after setting receive mode or write to ICDR after setting transmit mode.

16. Note on ACKE and TRS bits in slave mode

In the I²C bus interface, if 1 is received as the acknowledge bit value (ACKB = 1) in slave mode (TRS = 1) and then the address is received in slave mode without performing processing, interrupt handling may start at the rising edge of the 9th clock pulse even if the address does not match. Similarly, if the start condition or address is transmitted from master device in slave transmit mode (TRS = 1), the IRIC flag may be set after the start condition is set and 1 received as the acknowledge bit value (ACKB = 1), thus causing an interrupt source even when the address does not match.

To use the I²C bus interface module in slave mode, be sure to follow the procedures below.

- A. When having received 1 as the acknowledge bit value for the last transmit data at the end of a series of transmit operation, clear the ACKE bit in ICCR once to initialize the acknowledge bit to 0.
- B. Set receive mode (TRS = 0) before the next start condition is input in slave mode. Complete transmit operation by the procedure shown in figure 16.24, in order to switch from slave transmit mode to slave receive mode.

16.6.1 Module Stop Mode Setting

The IIC operation can be enabled or disabled using the module stop control register. The default setting is for the IIC operation to be halted. Register access is enabled by canceling module stop mode. For details, refer to section 26, Power-Down Modes.

17.1 Features

- Conforms to PS/2 interface specifications
- Direct bus drive (via the KCLK and KD pins)
- Interrupt sources: on completion of data reception and on detection of clock edge
- Error detection: parity error and stop bit monitoring

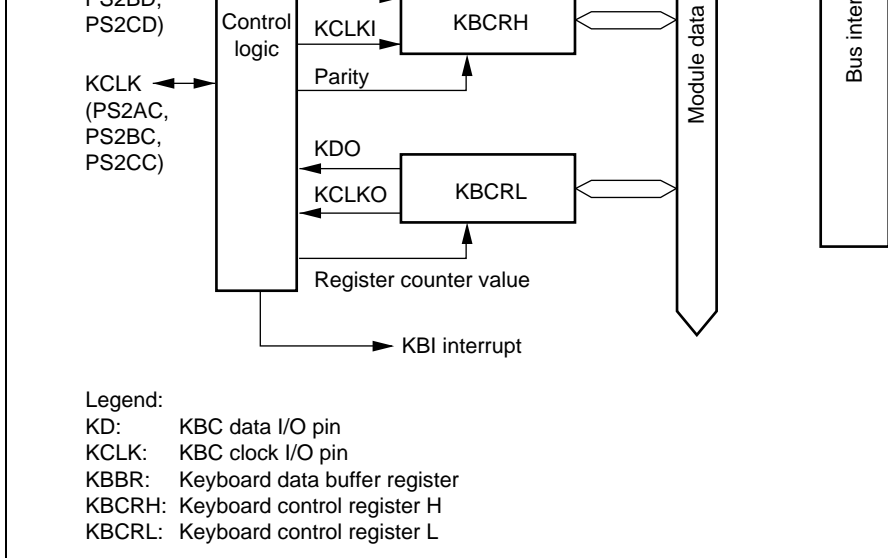


Figure 17.1 Block Diagram of Keyboard Buffer Controller

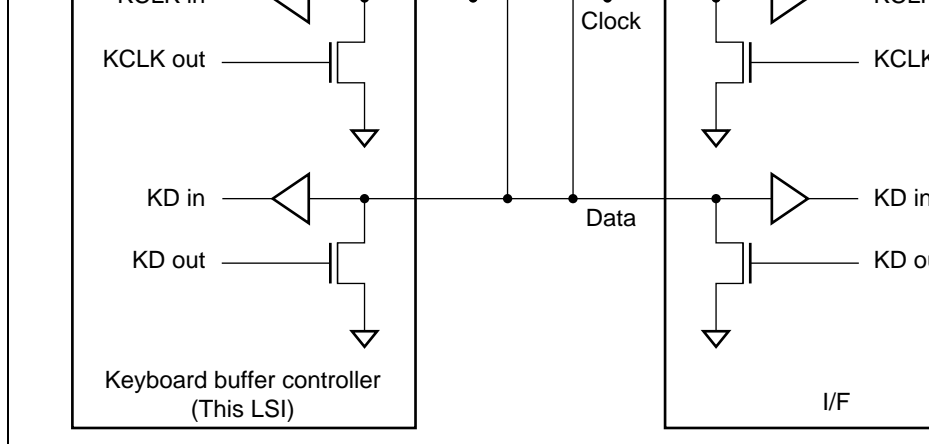


Figure 17.2 Keyboard Buffer Controller Connection

17.2 Input/Output Pins

Table 17.1 lists the input/output pins used by the keyboard buffer controller.

Table 17.1 Pin Configuration

Channel	Name	Abbreviation*	I/O	Function
0	KBC clock I/O pin (KCLK0)	PS2AC	I/O	KBC clock in
	KBC data I/O pin (KD0)	PS2AD	I/O	KBC data in
1	KBC clock I/O pin (KCLK1)	PS2BC	I/O	KBC clock in
	KBC data I/O pin (KD1)	PS2BD	I/O	KBC data in
2	KBC clock I/O pin (KCLK2)	PS2CC	I/O	KBC clock in
	KBC data I/O pin (KD2)	PS2CD	I/O	KBC data in

Note: * These are the external I/O pin names. In the text, clock I/O pins are referred to as KCLK and data I/O pins as KD, omitting the channel designations.

17.3.1 Keyboard Control Register H (KBCRH)

KBCRH indicates the operating status of the keyboard buffer controller.

Bit	Bit Name	Initial Value	R/W	Description
7	KBIOE	0	R/W	Keyboard In/Out Enable Selects whether or not the keyboard buffer controller is used. 0: The keyboard buffer controller is non-operational (KCLK and KD signal pins have port function disabled) 1: The keyboard buffer controller is enabled for transmission and reception (KCLK and KD signal pins are in the bus drive state)
6	KCLKI	1	R	Keyboard Clock In Monitors the KCLK I/O pin. This bit cannot be modified. 0: KCLK I/O pin is low 1: KCLK I/O pin is high
5	KDI	1	R	Keyboard Data In Monitors the KDI I/O pin. This bit cannot be modified. 0: KD I/O pin is low 1: KD I/O pin is high
4	KBFSEL	1	R/W	Keyboard Buffer Register Full Select Selects whether the KBF bit is used as the keyboard buffer register full flag or as the KCLK fall interrupt flag. When KBFSEL is cleared to 0, the KBF bit is used as the keyboard buffer register full flag. KBCRL should be cleared to 0 to disable reception. 0: KBF bit is used as KCLK fall interrupt flag 1: KBF bit is used as keyboard buffer register full flag

indicates that data reception has been completed and the received data is in KBBR.

0: [Clearing condition]

Read KBF when KBF =1, then write 0 in KBFSEL

1: [Setting conditions]

- When data has been received normally and has been transferred to KBBR while KBFS = 1 (keyboard buffer register full flag)
- When a KCLK falling edge is detected and KBFSEL = 0 (KCLK interrupt flag)

1	PER	0	R/(W)*	Parity Error
				Indicates that an odd parity error has occurred.
				0: [Clearing condition]
				Read PER when PER =1, then write 0 in PERSEL
				1: [Setting condition]
				When an odd parity error occurs

0	KBS	0	R	Keyboard Stop
				Indicates the receive data stop bit. Valid only when KBF = 1.
				0: 0 stop bit received
				1: 1 stop bit received

Note: * Only 0 can be written for clearing the flag.

				0: Loading of receive data into KBBR is disabled 1: Loading of receive data into KBBR is enabled
6	KCLKO	1	R/W	Keyboard Clock Out Controls KBC clock I/O pin output. 0: KBC clock I/O pin is low 1: KBC clock I/O pin is high
5	KDO	1	R/W	Keyboard Data Out Controls KBC data I/O pin output. 0: KBC data I/O pin is low 1: KBC data I/O pin is high
4	—	1	—	Reserved This bit is always read as 1 and cannot be written.

0000: —
 0001: Start bit
 0010: KB0
 0011: KB1
 0100: KB2
 0101: KB3
 0110: KB4
 0111: KB5
 1000: KB6
 1001: KB7
 1010: Parity bit
 1011: —
 11- - : —

17.3.3 Keyboard Data Buffer Register (KBBR)

KBBR stores receive data. Its value is valid only when KBF = 1.

Bit	Bit Name	Initial Value	R/W	Description
7	KB7	0	R	Keyboard Data 7 to 0
6	KB6	0	R	8-bit read only data.
5	KB5	0	R	Initialized to H'00 by a reset, in standby mode, subactive mode, subsleep mode, a
4	KB4	0	R	stop mode, and when KBIOE is cleared to
3	KB3	0	R	
2	KB2	0	R	
1	KB1	0	R	
0	KB0	0	R	

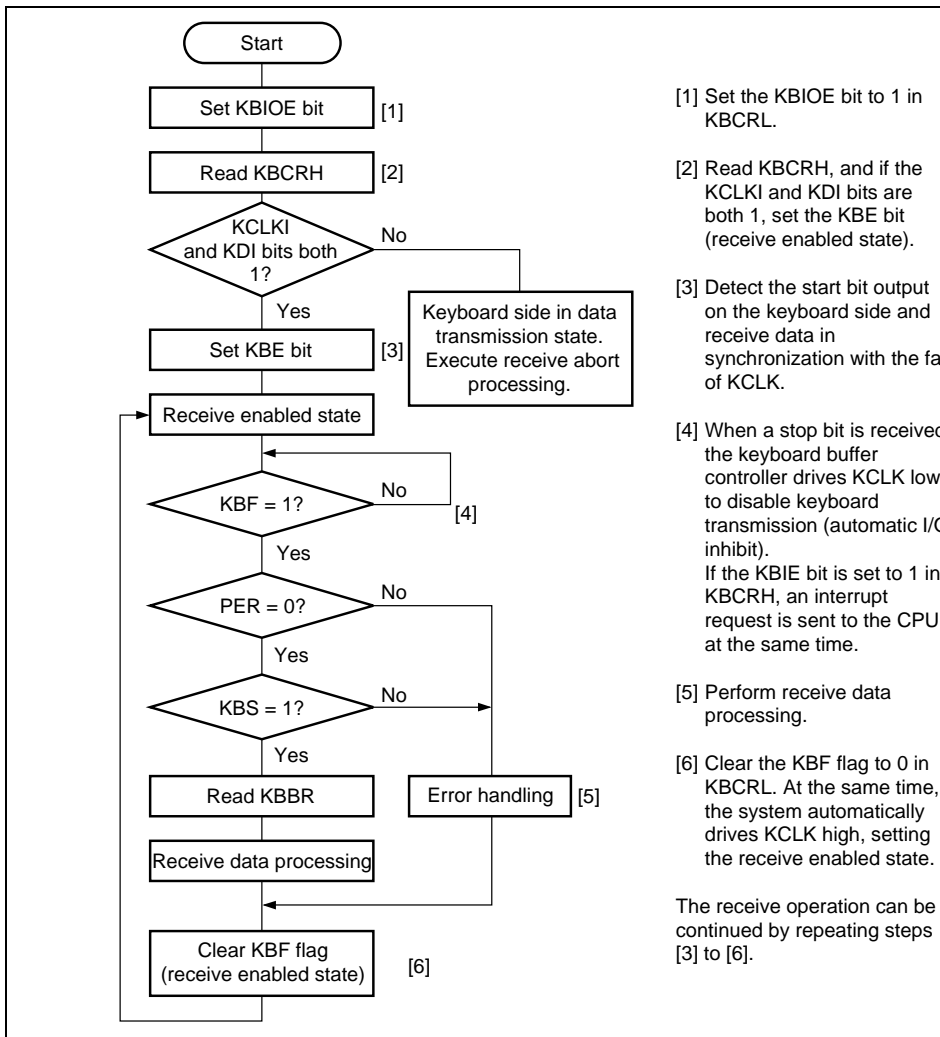


Figure 17.3 Sample Receive Processing Flowchart

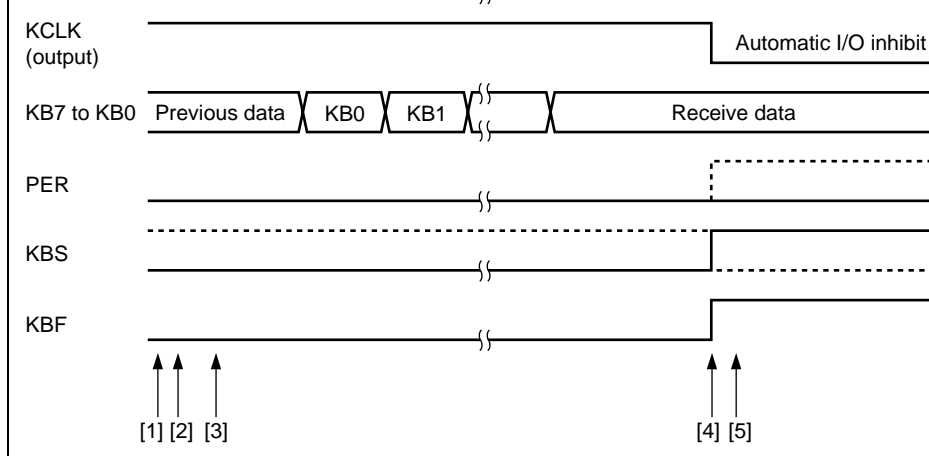


Figure 17.4 Receive Timing

17.4.2 Transmit Operation

In a transmit operation, KCLK (clock) is an output on the keyboard side, and KD (data) is an output on the chip (system) side. KD outputs a start bit, 8 data bits (LSB-first), an odd parity bit, and a stop bit, in that order. The KD value is valid when KCLK is high. A sample transmit processing flowchart is shown in figure 17.5, and the transmit timing is shown in figure 17.6.

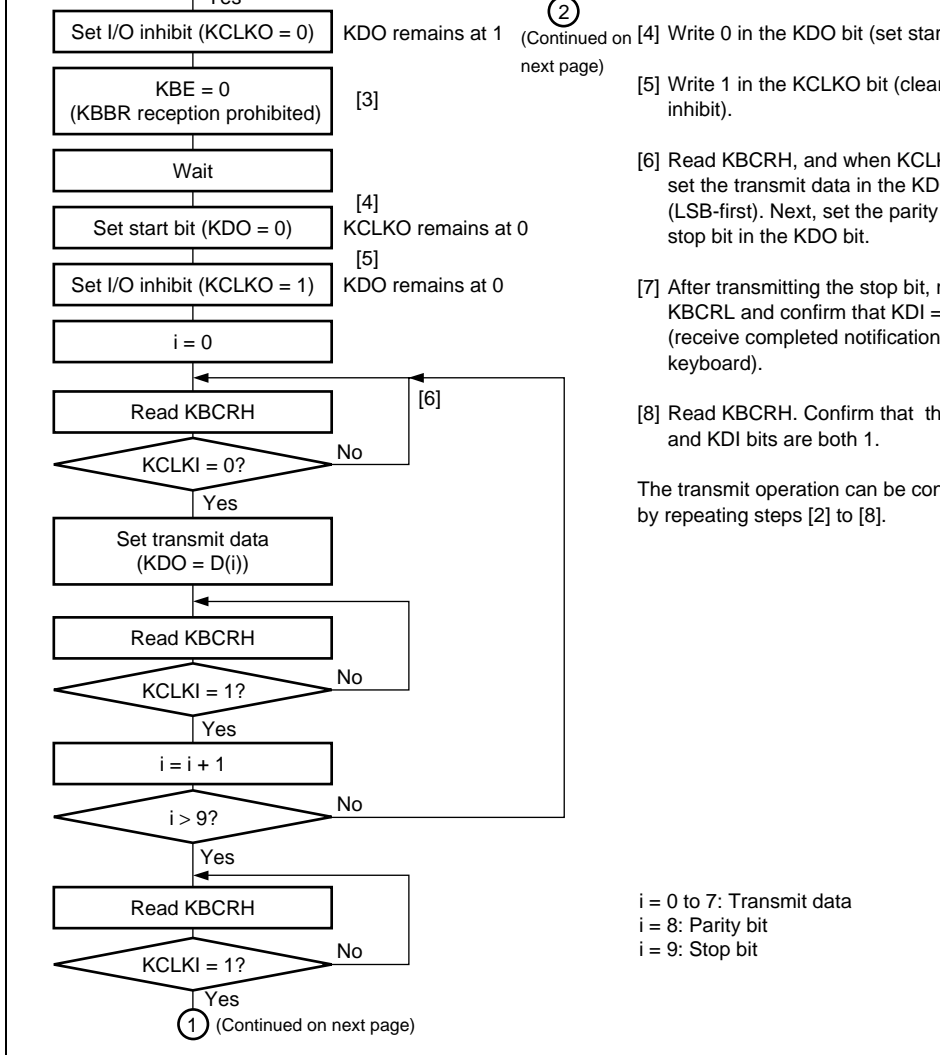


Figure 17.5 (1) Sample Transmit Processing Flowchart

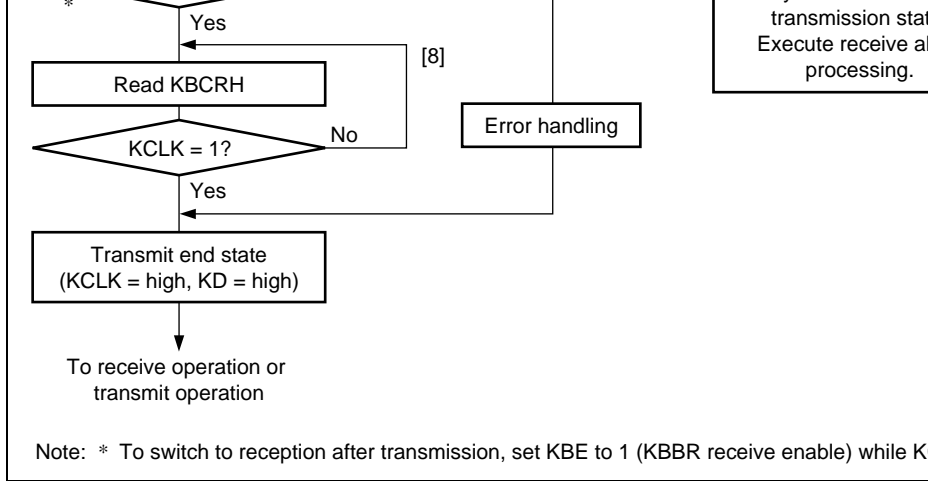


Figure 17.5 (2) Sample Transmit Processing Flowchart

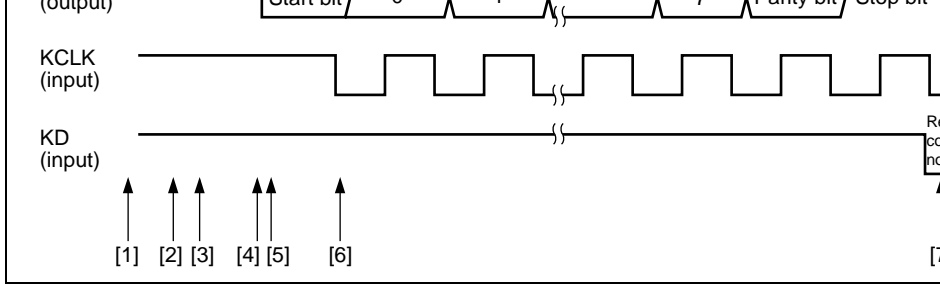


Figure 17.6 Transmit Timing

17.4.3 Receive Abort

This LSI (system side) can forcibly abort transmission from the device connected to it (device side) in the event of a protocol error, etc. In this case, the system holds the clock low. During reception, the keyboard also outputs a clock for synchronization, and the clock is monitored. If the keyboard output clock is high. If the clock is low at this time, the keyboard judges this as an abort request from the system, and data transmission from the keyboard is aborted. The system can abort reception by holding the clock low for a certain period. A sample receive processing flowchart is shown in figure 17.7, and the receive abort timing in figure 17.8.

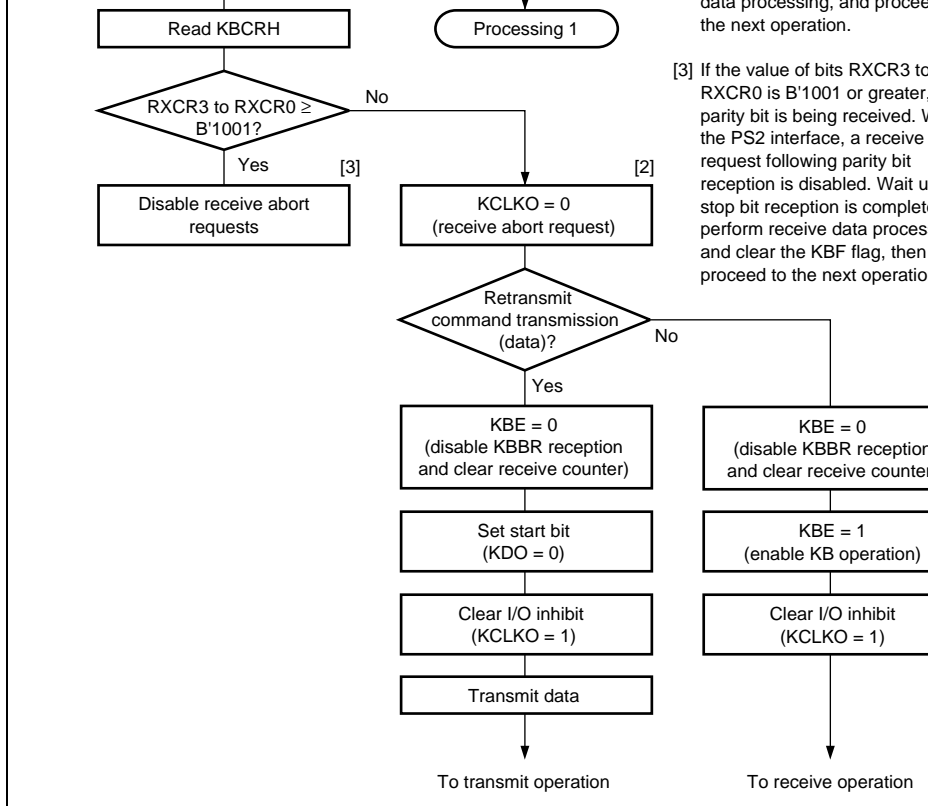


Figure 17.7 (1) Sample Receive Abort Processing Flowchart

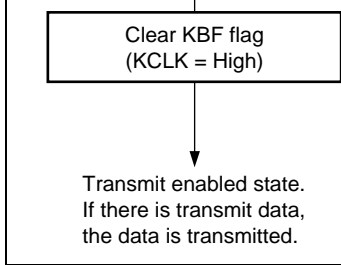


Figure 17.7 (2) Sample Receive Abort Processing Flowchart

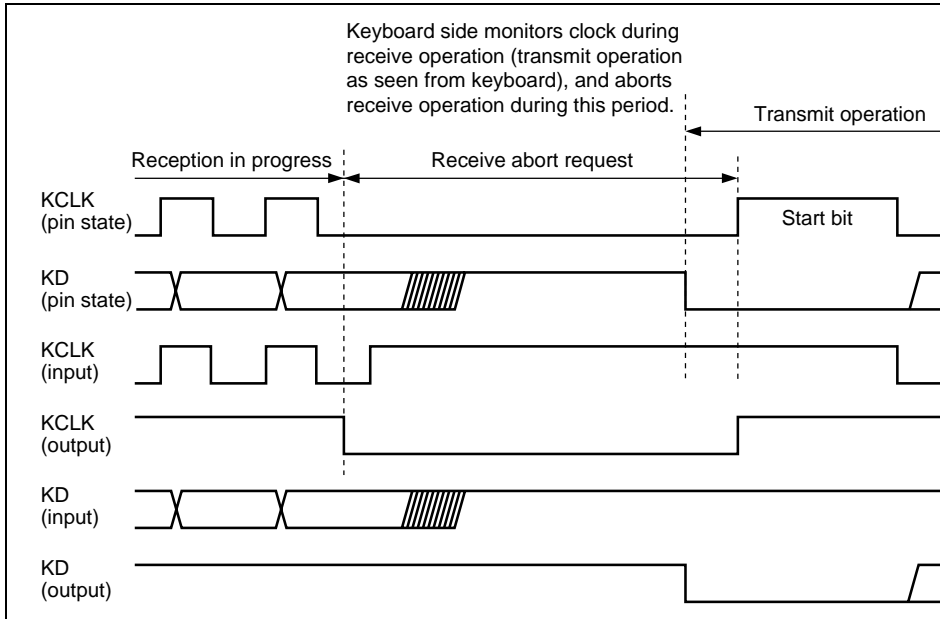


Figure 17.8 Receive Abort and Transmit Start (Transmission/Reception Switchover) Timing

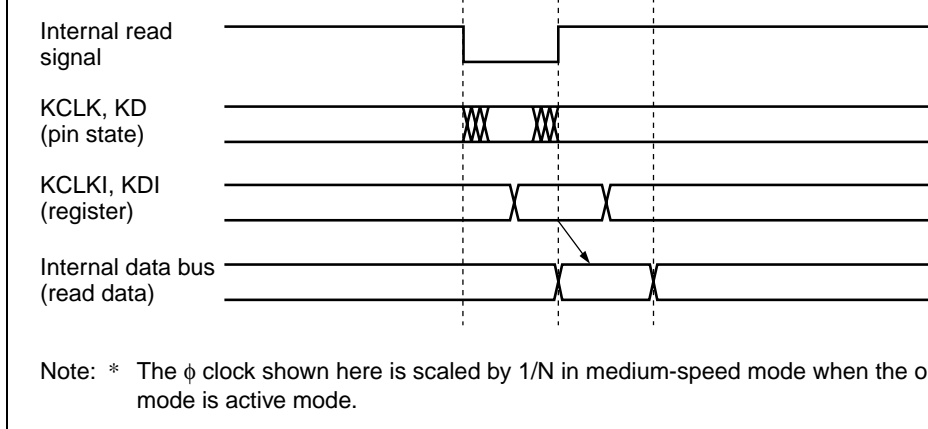


Figure 17.9 KCLKI and KDI Read Timing

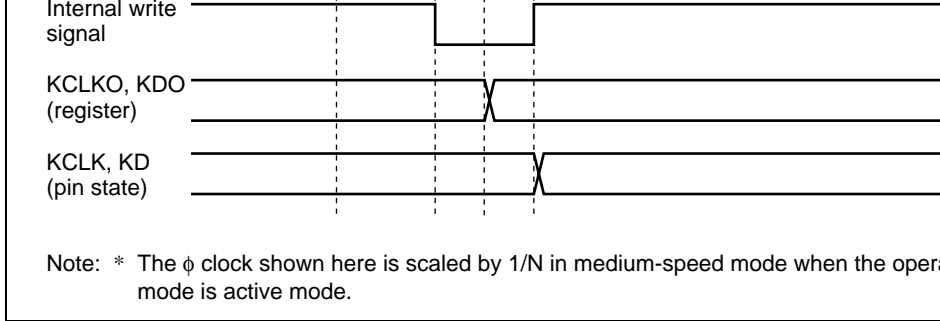


Figure 17.10 KCLKO and KDO Write Timing

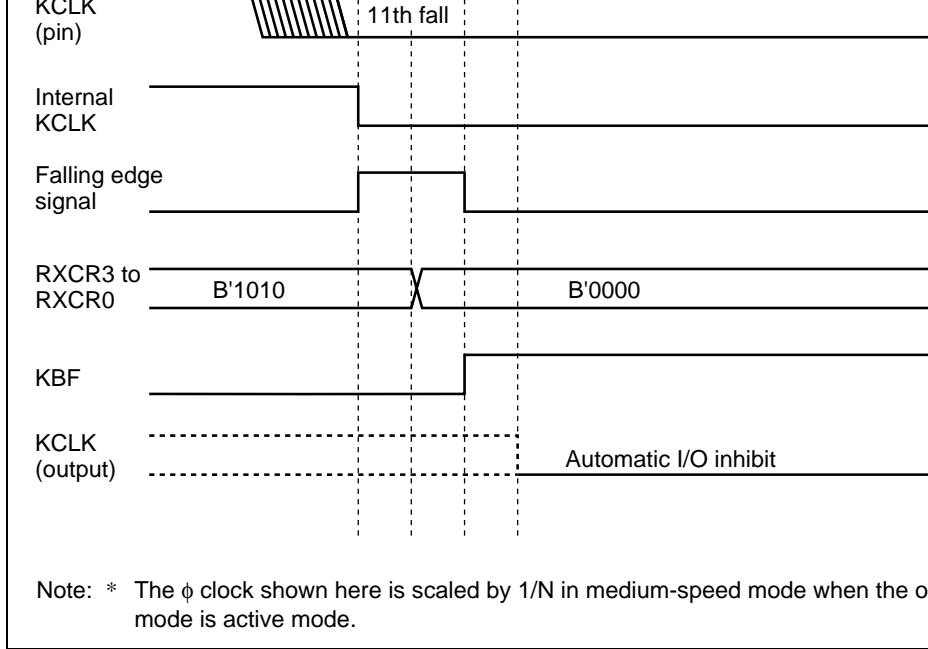


Figure 17.11 KBF Setting and KCLK Automatic I/O Inhibit Generation Timing

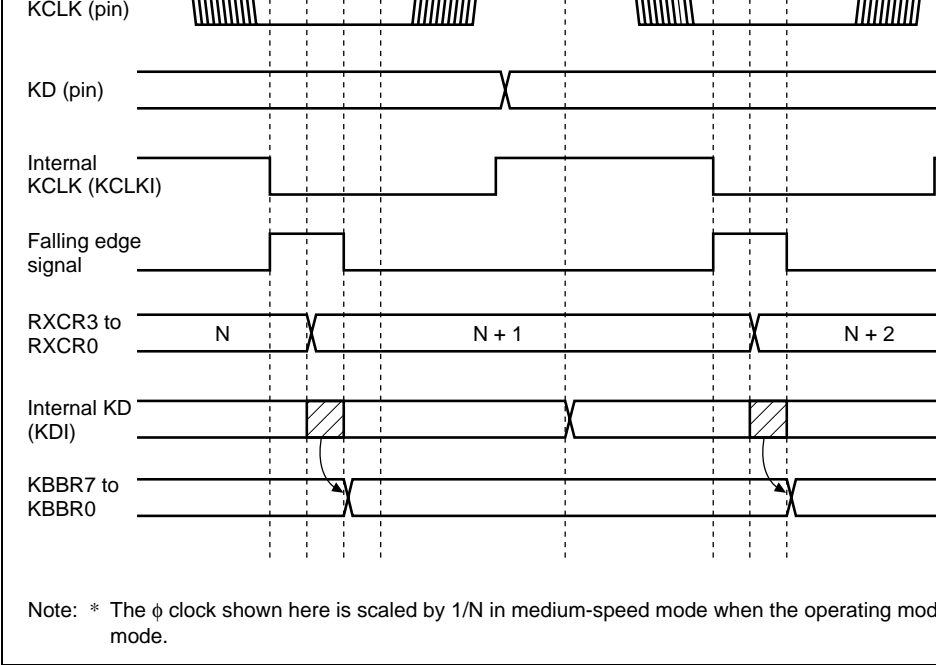
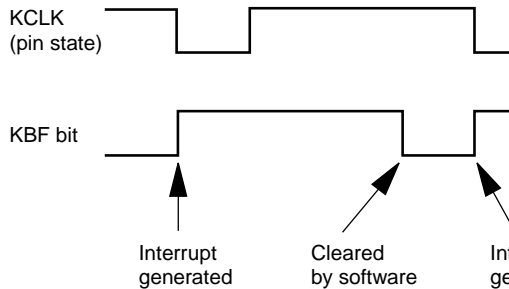
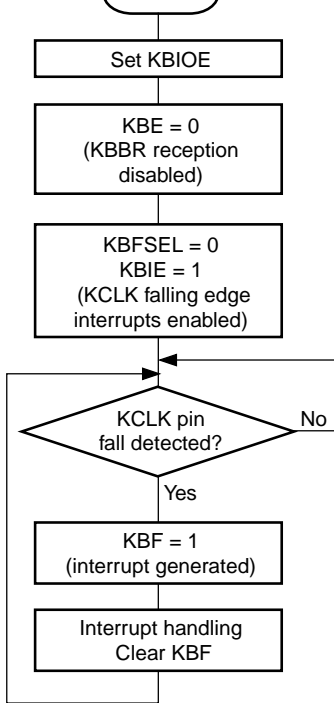


Figure 17.12 Receive Counter and KBBR Data Load Timing



Note: * The KBF setting timing is the same as the timing of KBF setting and KCLK automatic I/O generation in figure 17.11. When the KBF bit is used as the KCLK input fall interrupt flag, automatic I/O inhibit function does not operate.

Figure 17.13 Example of KCLK Input Fall Interrupt Operation

If the KBFSEL bit and KBE bit are both 0 at this time, the KBF bit is set. Figure 17.14 shows the timing of KBIOE setting and KCLK falling edge detection.

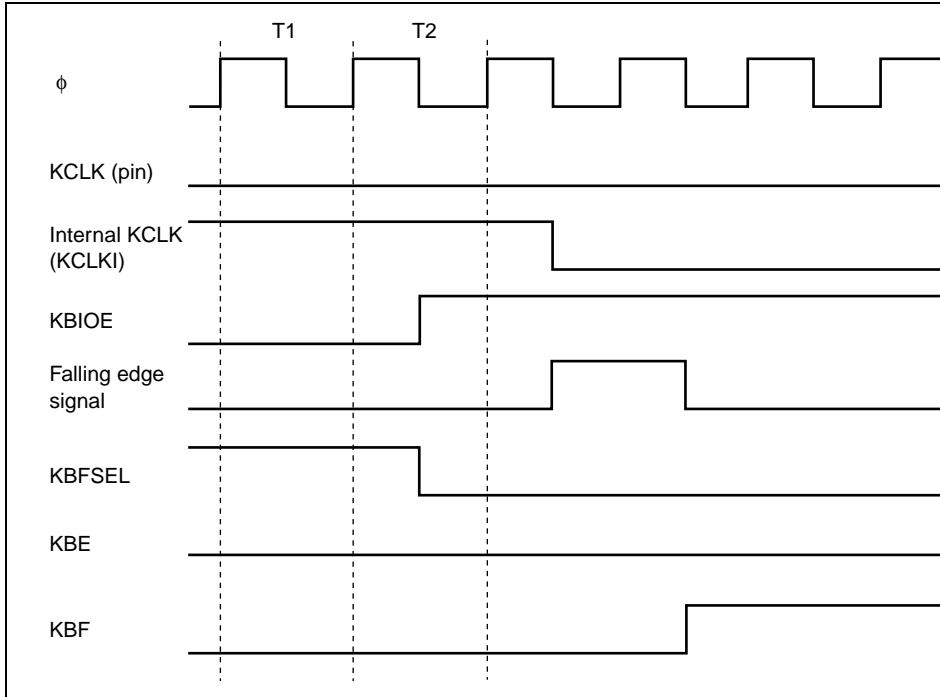


Figure 17.14 KBIOE Setting and KCLK Falling Edge Detection Timing

17.5.2 Module Stop Mode Setting

Keyboard buffer controller operation can be enabled or disabled using the module stop register. The initial setting is for keyboard buffer controller operation to be halted. Register is enabled by canceling module stop mode. For details, refer to section 26, Power-Down

Communication is carried out via seven control signals from the host processor ($\overline{CS1}$, $\overline{ECS2}$, $\overline{CS3}$, $\overline{CS4}$, $\overline{HA0}$, \overline{IOR} , and \overline{IOW}), six output signals to the host processor ($\overline{GA2}$, $\overline{HIRQ11}$, $\overline{HIRQ12}$, $\overline{HIRQ3}$, and $\overline{HIRQ4}$), and an 8-bit bidirectional command/data bus ($\overline{HDB0}$). The $\overline{CS1}$, $\overline{CS2}$ (or $\overline{ECS2}$), $\overline{CS3}$ and $\overline{CS4}$ signals select one of the four interfaces.

18.1 Features

- Control of the fast GATE A20 function
- Shutdown of the XBS module by the HIFSD pin
- Five host interrupt requests

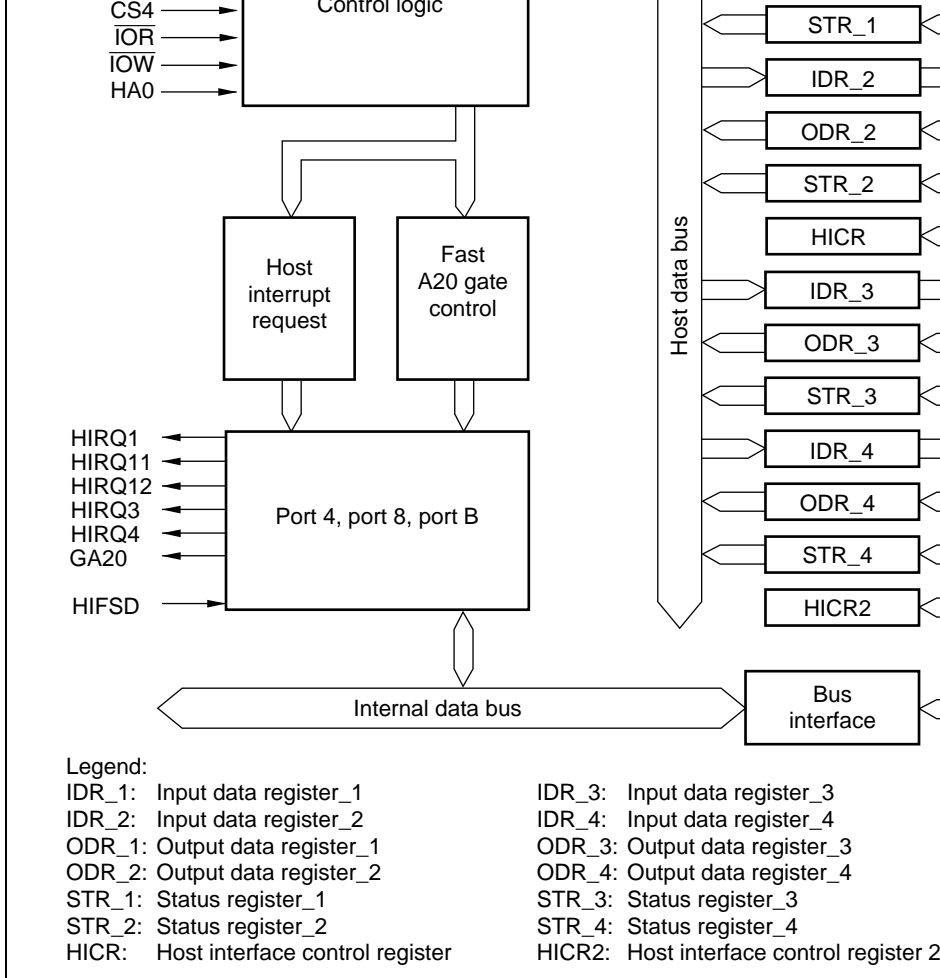


Figure 18.1 Block Diagram of XBS

Chip select 1	$\overline{CS1}$	P95	Input	Host interface chip select signal ODR_1, STR_1
Chip select 2*	$\overline{CS2}$	P81	Input	Host interface chip select signal ODR_2, STR_2
	$\overline{ECS2}$	P90		
Chip select 3	$\overline{CS3}$	PB2	Input	Host interface chip select signal ODR_3, STR_3
Chip select 4	$\overline{CS4}$	PB3	Input	Host interface chip select signal ODR_4, STR_4
Command/data	HA0	P80	Input	Host interface address select signal In host read access, this signal is used to select the status registers (STR_1 to STR_4) and the output data registers (ODR_1 to ODR_4). In host write access to the data registers (IDR_1 to IDR_4), this signal indicates whether the host is writing a command or data.
Data bus	HDB7 to HDB0	P37 to P30	I/O	Host interface data bus
Host interrupt 11	HIRQ11	P43	Output	Interrupt output 11 to host
Host interrupt 1	HIRQ1	P44	Output	Interrupt output 1 to host
Host interrupt 12	HIRQ12	P45	Output	Interrupt output 12 to host
Host interrupt 3	HIRQ3	PB0	Output	Interrupt output 3 to host
Host interrupt 4	HIRQ4	PB1	Output	Interrupt output 4 to host
Gate A20	GA20	P81	Output	A20 gate control signal output
HIF shutdown	HIFSD	P82	Input	Host interface shutdown control signal

Note: * Selection of $\overline{CS2}$ or $\overline{ECS2}$ is by means of the CS2E bit in STCR and the FGA20E bit in HICR. XBS channel 2 and the $\overline{CS2}$ pin can be used when CS2E = 1. When $\overline{CS2}$ is used when FGA20E = 0, and $\overline{ECS2}$ is used when FGA20E = 1. In this case, both are referred to as $\overline{CS2}$.

- Host interface control register 2 (HICR2)
- Input data register (IDR)
- Output data register (ODR)
- Status register (STR)

18.3.1 System Control Register 2 (SYSCR2)

SYSCR2 controls the operations of port 6 and host interface.

Bit	Bit Name	Initial Value	R/W	Description
7	KWUL1	0	R/W	Key Wakeup Level 1 and 0
6	KWUL0	0	R/W	Sets the port 6 input level. The input level of multiplexing pins is also changed by these bits. 00: Port 6 is in the standard input level 01: Port 6 is in input level 1 10: Port 6 is in input level 2 11: Port 6 is in input level 3
5	P6PUE	0	R/W	Port 6 Input Pull-Up MOS Extra (P6PUE) Controls and selects the current specification of port 6 input pull-up MOS. 0: Standard current specification 1: Current limited specification
4	—	0	—	Reserved Only 0 should be written to this bit.

2	CS4E	0	R/W	CS4 Enable 0: Channel 4 functions disabled 1: Channel 4 functions enabled (channel 4 enabled) Enabling setting is valid when the HI12E b
1	CS3E	0	R/W	CS3 Enable 0: Channel 3 functions disabled 1: Channel 3 functions enabled (channel 3 enabled) Enabling setting is valid when the HI12E b
0	HI12E	0	R/W	Host Interface Enable Bit 0: Host interface functions are disabled 1: Host interface functions are enabled (se CS2E to CS4E, FGA20E, and SDE are Enabling setting is valid in single-chip mod

Bit	Bit Name	Initial Value	Slave	Host	Description
7 to 3	—	All 1	—	—	Reserved These bits are always read as 1 and cannot be modified.
2	IBFIE2	0	R/W	—	Input Data Register Full Interrupt Enable Enables or disables the IBF2 interrupt request to the internal CPU. 0: Input data register (IDR_2) reception completed interrupt request disabled 1: Input data register (IDR_2) reception completed interrupt request enabled
1	IBFIE1	0	R/W	—	Input Data Register Full Interrupt Enable Enables or disables the IBF1 interrupt request to the internal CPU. 0: Input data register (IDR_1) reception completed interrupt request disabled 1: Input data register (IDR_1) reception completed interrupt request enabled

0: XBS fast A20 gate function disabled

1: XBS fast A20 gate function enabled

When the fast A20 gate is disabled, the fast A20 gate can be implemented by the operation of the P81 output.

When the host interface (XBS) fast A20 gate function is enabled, the DDR bit for P81 is set to 1. Therefore, the state of the P81 cannot be monitored by reading the P81.

A fast A20 gate function is also provided by the LPC. The state of the P81/GA20 pin can be monitored by reading the LPC's GA20 register.

					Enables or disables the IBF4 interrupt internal CPU.
					0: Input data register (IDR_4) reception completed interrupt request disabled
					1: Input data register (IDR_4) reception completed interrupt request enabled
1	IBFIE3	0	R/W	—	Input Data Register Full Interrupt Enable Enables or disables the IBF3 interrupt internal CPU.
					0: Input data register (IDR_3) reception completed interrupt request disabled
					1: Input data register (IDR_3) reception completed interrupt request enabled
0	—	0	—	—	Reserved The initial value should not be changed

6	IDR6	—	R	W	most data bus is written into IDR_n on the falling edge of \overline{IOW} . The HA0 state is also latched into the C/D bit in STR_n to indicate whether the written information is a command or data.
5	IDR5	—	R	W	
4	IDR4	—	R	W	
3	IDR3	—	R	W	
2	IDR2	—	R	W	
1	IDR1	—	R	W	
0	IDR0	—	R	W	

18.3.4 Output Data Register 1 (ODR)

ODR is a register in which data to be output from the slave processor (this LSI) to the master processor is stored.

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	ODR7	—	R/W	R	The ODR_n contents are output on the master data bus when HA0 is low, \overline{CSn} (n = 1 to 7) is low, and \overline{IOR} is low.
6	ODR6	—	R/W	R	
5	ODR5	—	R/W	R	
4	ODR4	—	R/W	R	
3	ODR3	—	R/W	R	
2	ODR2	—	R/W	R	
1	ODR1	—	R/W	R	
0	ODR0	—	R/W	R	

3	C/D	0	R	R	Command/Data Receives the HA0 input when the host processor writes to IDR, and indicates whether the register contains data or a command. 0: Contents of input data register (IDR) 1: Contents of input data register (IDR) command
2	DBU	0	R/W	R	Defined by User The user can use these bits as necessary.
1	IBF	0	R	R	Input Buffer Full This bit is an internal interrupt source for the slave processor (this LSI). The IBF flag setting and clearing conditions are different when the fast A20 gate is used. For details see table 18.5. [Clearing Condition] 0: When the slave processor reads IDR [Setting Condition] 1: When the host processor writes to IDR
0	OBF	0	R/(W)*	R	Output Buffer Full [Clearing Condition] 0: When the host processor reads OBF or the slave writes 0 in the OBF bit [Setting Condition] 1: When the slave processor writes to OBF

Note: * Only 0 can be written, to clear the flag.

OBF	Falling edge of slave's internal write signal when writing to ODR1	Rising edge of host's read signal (falling edge) when reading ODR1
-----	--	--

Note: * The IBF flag setting and clearing conditions are different when the fast A20 is used. For details see table 18.5.

18.4 Operation

18.4.1 Host Interface Activation

The host interface is activated by setting the HI12E bit in SYSCR2 to 1 in single-chip mode. When the host interface is activated, all related I/O ports (data port 3, control ports 8 and 9, host interrupt request port 4) become dedicated host interface ports. Setting the CS3E and CS4E bit to 1 enables the number of host interface channels to be extended to four, and channel 3 and 4 related I/O port (part of port B for control and host interrupt requests) becomes a host interface port.

			1	Host interface channel 1 and 4 functions operating Operation of channels 2 and 3 halted Pins P43, P81, P90, and PB0 to PB3 operate as I/O ports. $\overline{CS2}$ and $\overline{CS3}$ inputs do not operate.
	1	0	0	Host interface channel 1 and 3 functions operating Operation of channels 2 and 4 halted Pins P43, P81, P90, PB1, and PB3 operate as I/O ports. $\overline{CS2}$ and $\overline{CS4}$ inputs do not operate.
			1	Host interface channel 1, 3, and 4 functions operating Operation of channel 2 halted Pins P43, P81, and P90 operate as I/O ports. $\overline{CS2}$ input does not operate.
1	0	0	0	Host interface channel 1 and 2 functions operating Operation of channels 3 and 4 halted Pins PB0 to PB3 operate as I/O ports. $\overline{CS3}$ and $\overline{CS4}$ inputs do not operate.
			1	Host interface channel 1, 2, and 4 functions operating Operation of channel 3 halted Pins PB0 and PB2 operate as I/O ports. $\overline{CS3}$ input does not operate.
	1	0	0	Host interface channel 1 to 3 functions operating Operation of channel 4 halted Pins PB1 and PB3 operate as I/O ports. $\overline{CS4}$ input does not operate.
			1	Host interface channel 1 to 4 functions operating

		1	Setting prohibited
	1	0	Data read from output data register n
		1	Status read from status register n (ST
1	0	0	Data written to input data register n (I
		1	Command written to input data regist
	1	0	Idle state
		1	Idle state

Note: n = 1 to 4

18.4.3 A20 Gate

The A20 gate signal can mask address A20 to emulate an addressing mode used by personal computers with an 8086*-family CPU. A regular-speed A20 gate signal can be output under firmware control. Fast A20 gate output is enabled by setting the FGA20E bit (bit 0) to 1 (H'FFF0).

Note: * Intel microprocessor.

Regular A20 Gate Operation: Output of the A20 gate signal can be controlled by an A20 gate command followed by data. When the slave processor (this LSI) receives data, it normally outputs the data. When an interrupt routine activated by the IBF1 interrupt to read IDR1. If the data follows an A20 gate command, software copies bit 1 of the data and outputs it at the gate A20 pin.

Fast A20 Gate Operation: When the FGA20E bit is set to 1, P81/GA20 is used for output of the fast A20 gate signal. Bit P81DDR must be set to 1 to assign this pin for output. When P81DDR for P81 is set to 1, the state of the P81/GA20 pin cannot be monitored by reading the P81 register. The state of the P81/GA20 pin can be monitored by reading the GA20 bit in the HICR2 register. The initial output from this pin will be a logic 1, which is the initial value of the GA20 bit. Afterward, the host processor can manipulate the output from this pin by sending commands to the slave processor. This function is available only when register IDR1 is accessed using $\overline{CS1}$. The slave processor (this LSI) decodes the commands input from the host processor. When an H

Rev. 3.00 Mar 21, 2006 pag

REJ03

RENESAS

GA20
(P81)

Rising edge of the host's write signal (IOW) when bit 1 of the written data is 1 and the data follows an H'D1 host command

Rising edge of the host's write signal (IOW) when bit 1 of the written data is 1 and the data follows an H'D1 host command

Also, when bit FGA20E in HICR is set to 0

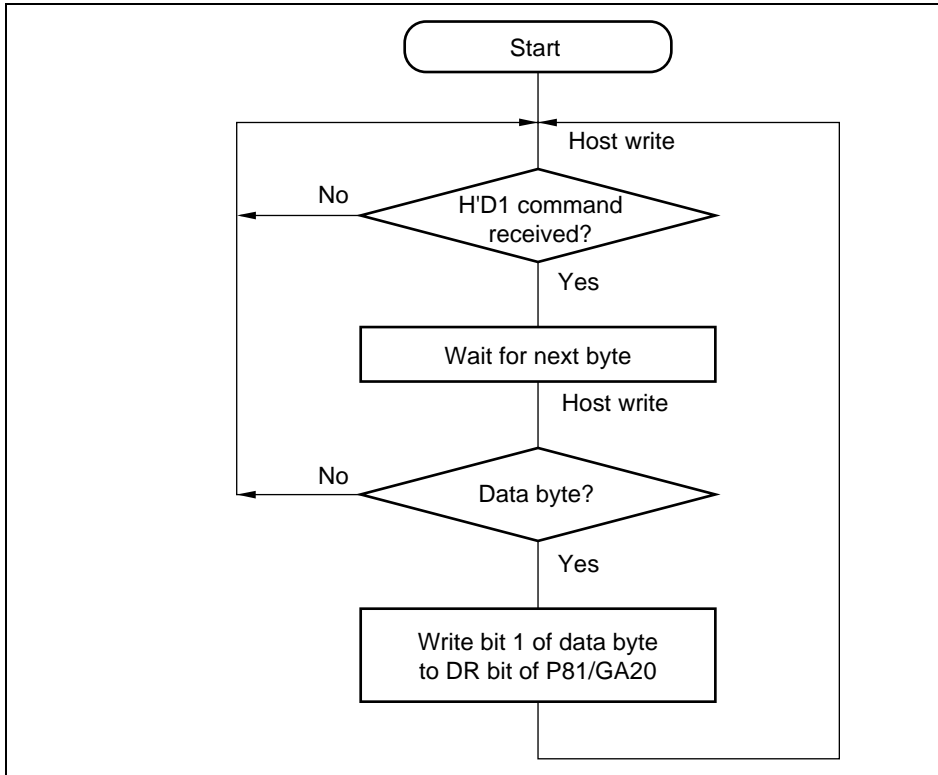


Figure 18.2 GA20 Output

0	0 data ^{*2}	0	0	
1	H'FF command	0	Q (0)	
1	H'D1 command	0	Q	Turn-on sequence
0	1 data ^{*1}	0	1	(abbreviated for)
1/0	Command other than H'FF and H'D1	1	Q (1)	
1	H'D1 command	0	Q	Turn-off sequence
0	0 data ^{*2}	0	0	(abbreviated for)
1/0	Command other than H'FF and H'D1	1	Q (0)	
1	H'D1 command	0	Q	Cancelled sequence
1	Command other than H'D1	1	Q	
1	H'D1 command	0	Q	Retriggered sequence
1	H'D1 command	0	Q	
1	H'D1 command	0	Q	Consecutively
0	Any data	0	1/0	sequences
1	H'D1 command	0	Q (1/0)	

Notes: 1. Arbitrary data with bit 1 set to 1.
2. Arbitrary data with bit 1 cleared to 0.

18.4.4 Host Interface Pin Shutdown Function

Host interface output can be placed in the high-impedance state according to the state of the HIFSD pin. Setting the SDE bit to 1 in the SYSCR2 register when the HI12E bit is set enables the HIFSD pin. The HIF constantly monitors the HIFSD pin, and when this pin goes low, the host interface output pins (HIRQ1, HIRQ11, HIRQ12, HIRQ3, HIRQ4, and GA20) go to the high-impedance state. At the same time, the host interface input pins ($\overline{CS1}$, $\overline{CS2}$ or \overline{EC} , $\overline{CS4}$, \overline{IOW} , \overline{IOR} , and HA0) are disabled (fixed at the high input state internally) regardless of pin states, and the signals of the multiplexed functions of these pins (input block) are also fixed internally. As a result, the host interface I/O pins (HDB7 to HDB0) also go to the high-impedance state.

Abbreviation	Pin	Drive mode	I/O	Selection conditions
$\overline{\text{IOR}}$	P93	O	Input	HI12E = 1
$\overline{\text{IOW}}$	P94	O	Input	HI12E = 1
$\overline{\text{CS1}}$	P95	O	Input	HI12E = 1
$\overline{\text{CS2}}$	P81	Δ	Input	HI12E = 1 and CS2E = 1 and FGA
$\overline{\text{ECS2}}$	P90	Δ	Input	HI12E = 1 and CS2E = 1 and FGA
$\overline{\text{CS3}}$	PB2	Δ	Input	HI12E = 1 and CS3E = 1
$\overline{\text{CS4}}$	PB3	Δ	Input	HI12E = 1 and CS4E = 1
HA0	P80	O	Input	HI12E = 1
HDB7 to HDB0	P37 to P30	O	I/O	HI12E = 1
HIRQ11	P43	Δ	Output	HI12E = 1 and CS2E = 1 and P43D
HIRQ1	P44	Δ	Output	HI12E = 1 and P44DDR = 1
HIRQ12	P45	Δ	Output	HI12E = 1 and P45DDR = 1
HIRQ3	PB0	Δ	Output	HI12E = 1 and CS3E = 1 and PB0D
HIRQ4	PB1	Δ	Output	HI12E = 1 and CS4E = 1 and PB1D
GA20	P81	Δ	Output	HI12E = 1 and FGA20E = 1
HIFSD	P82	—	Input	HI12E = 1 and SDE = 1

Legend:

O: Pins shut down by shutdown function

The $\overline{\text{IRQ2/ADTRG}}$ input signal is also fixed in the case of P90 shutdown, the $\overline{\text{TMCI}}$ signal in the case of P43 shutdown, and the $\overline{\text{TMRI/CSYNCI}}$ in the case of P45 shutdown.

Δ : Pins shut down only when the XBS function is selected by means of a register setting.

—: Pin not shut down

Table 18.8 Input Buffer Full Interrupts

Interrupt	Description
IBF1	Requested when IBFIE1 is set to 1 and IDR_1 is full
IBF2	Requested when IBFIE2 is set to 1 and IDR_2 is full
IBF3	Requested when IBFIE3 is set to 1 and IDR_3 is full
IBF4	Requested when IBFIE4 is set to 1 and IDR_4 is full

18.5.2 HIRQ11, HIRQ1, HIRQ12, HIRQ3, and HIRQ4

Bits P45DR to P43DR in the port 4 data register (P4DR) and bits PB1ODR and PB0ODR in the port B data register (PBODR) can be used as host interrupt request latches. When they are set to 1, they indicate a host interrupt request output, set each bit in the data direction register (DDR) of the port.

The corresponding bits in P4DR are cleared to 0 by the host processor's read signal (\overline{IO}) and HA0 are low, when \overline{IOR} goes low and the host reads ODR_1, HIRQ1 and HIRQ12 are cleared to 0. If $\overline{CS2}$ and HA0 are low, when \overline{IOR} goes low and the host reads ODR_2, HIRQ2 and HIRQ11 are cleared to 0. The corresponding bit in PBODR is cleared to 0 by the host's read signal (\overline{IOR}) and HA0 are low, when \overline{IOR} goes low and the host reads ODR_3, HIRQ3 is cleared to 0. If $\overline{CS2}$ and HA0 are low, when \overline{IOR} goes low and the host reads ODR_4, HIRQ4 is cleared to 0. When a host interrupt request, normally on-chip firmware writes 1 in the corresponding bit. When the interrupt, the host's interrupt handling routine reads the output data register (ODR_1, ODR_2, ODR_3, or ODR_4) and this clears the host interrupt latch to 0.

Table 18.9 indicates how these bits are set and cleared. Figure 18.3 shows the process flowchart form.

HIRQ12 (P45)	Internal CPU reads 0 from bit P45DR, then writes 1	(ODR_1) Internal CPU writes 0 in bit host reads output data regi (ODR_1)
HIRQ3 (PB0)	Internal CPU reads 0 from bit PB0ODR, then writes 1	Internal CPU writes 0 in bit or host reads output data r (ODR_3)
HIRQ4 (PB1)	Internal CPU reads 0 from bit PB1ODR, then writes 1	Internal CPU writes 0 in bit or host reads output data r (ODR_4)

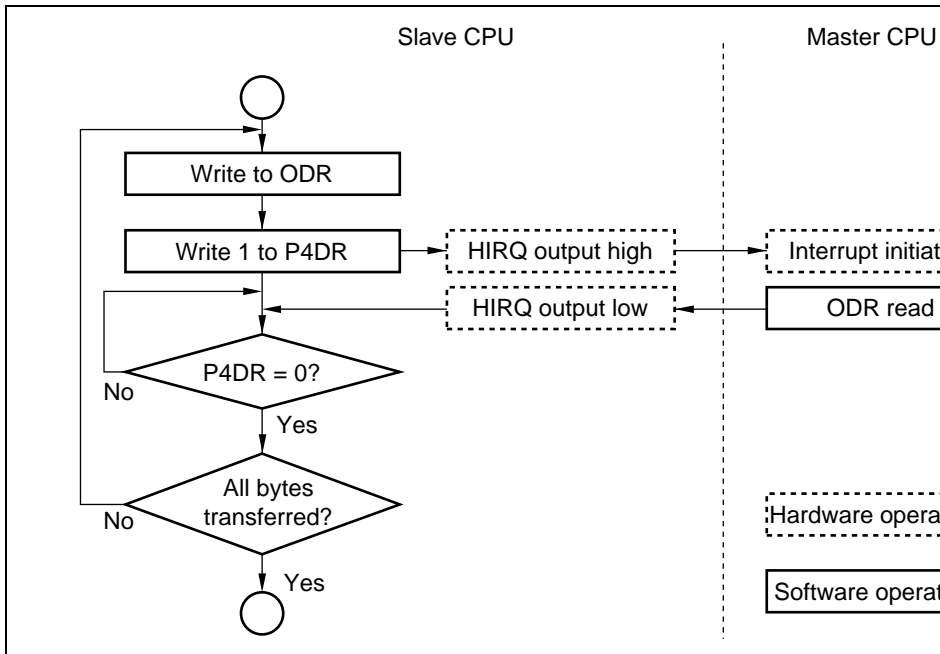


Figure 18.3 HIRQ Output Flowchart (Example of Channels 1 and 2)

The host interface provides buffering of asynchronous data from the host processor and processor (this LSI), but an interface protocol must be followed to implement necessary and avoid data contention. For example, if the host and slave processors try to access the input or output data register simultaneously, the data will be corrupted. Interrupts can design a simple and effective protocol.

Also, if two or more of pins $\overline{CS1}$ to $\overline{CS4}$ are driven low simultaneously in attempting access, signal contention will occur within the chip, and a through-current may result. must therefore be avoided.

18.6.2 Module Stop Mode Setting

XBS operation can be enabled or disabled using the module stop control register. The setting is for XBS operation to be halted. Register access is enabled by canceling mode. For details, refer to section 26, Power-Down Modes.

It is also provided with power-down functions that can control the PCI clock and shut down the host interface.

19.1 Features

- Supports LPC interface I/O read cycles and I/O write cycles
 - Uses four signal lines (LAD3 to LAD0) to transfer the cycle type, address, and data.
 - Uses three control signals: clock (LCLK), reset ($\overline{\text{LRESET}}$), and frame ($\overline{\text{LFRAM}}$).
- Has three register sets comprising data and status registers
 - The basic register set comprises three bytes: an input register (IDR), output register (ODR), and status register (STR).
 - Channels 1 and 2 have fixed I/O addresses of H'60/H'64 and H'62/H'66, respectively. An A20 gate function is also provided.
 - The I/O address can be set for channel 3. Sixteen bidirectional data register bytes can be manipulated in addition to the basic register set.
- Supports SERIRQ
 - Host interrupt requests are transferred serially on a single signal line (SERIRQ).
 - On channel 1, HIRQ1 and HIRQ12 can be generated.
 - On channels 2 and 3, SMI, HIRQ6, and HIRQ9 to HIRQ11 can be generated.
 - Operation can be switched between quiet mode and continuous mode.
 - The $\overline{\text{CLKRUN}}$ signal can be manipulated to restart the PCI clock (LCLK).
- Eleven interrupt sources
 - The LPC module can be shut down by inputting the $\overline{\text{LPCPD}}$ signal.
 - Three pins, $\overline{\text{PME}}$, $\overline{\text{LSMI}}$, and LSCI, are provided for general input/output.

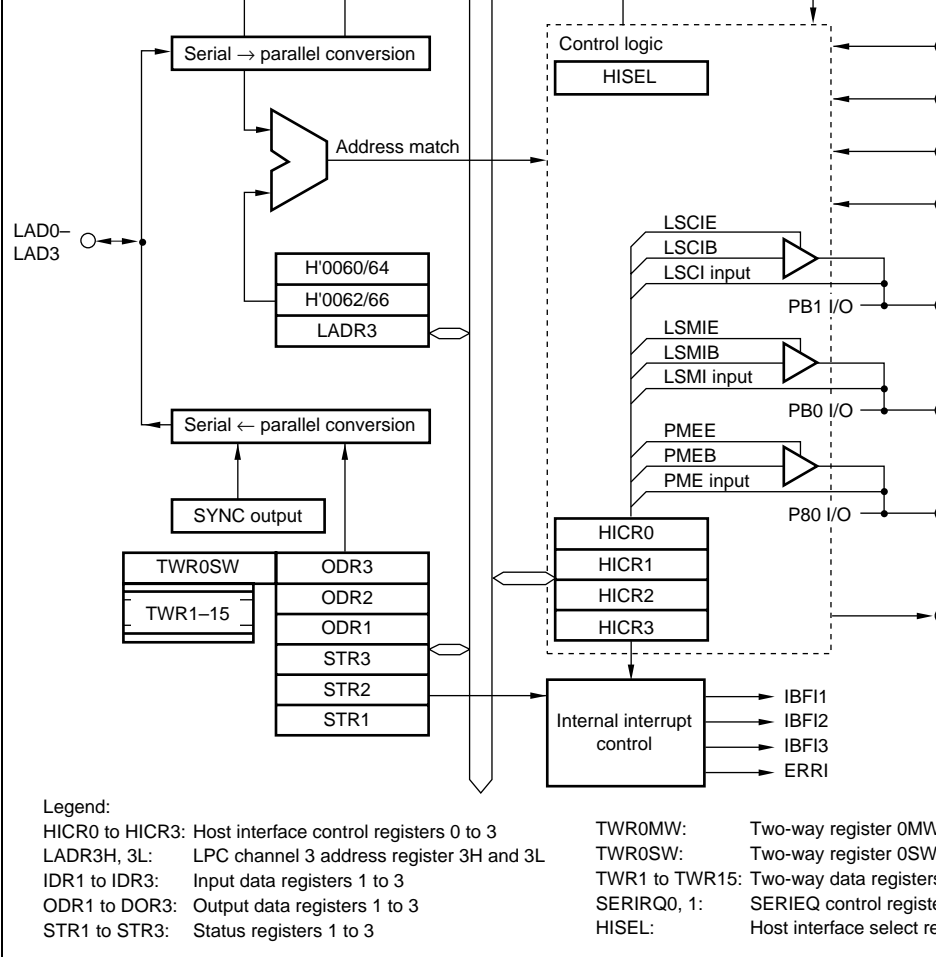


Figure 19.1 Block Diagram of LPC

data 0 to 0			output	type/address/data signal synchronized with LCLK
LPC frame	$\overline{\text{LFRAME}}$	P34	Input ^{*1}	Transfer cycle start and termination signal
LPC reset	$\overline{\text{LRESET}}$	P35	Input ^{*1}	LPC interface reset signal
LPC clock	LCLK	P36	Input	33 MHz PCI clock signal
Serialized interrupt request	SERIRQ	P37	Input/output ^{*1}	Serialized host interrupt signal, synchronized with LCLK (SMI, IRQ1, IRQ6, IRQ7)
LSCI general output	LSCI	PB1	Output ^{*1 *2}	General output
LSMI general output	$\overline{\text{LSMI}}$	PB0	Output ^{*1 *2}	General output
PME general output	$\overline{\text{PME}}$	P80	Output ^{*1 *2}	General output
GATE A20	GA20	P81	Output ^{*1 *2}	A20 gate control signal
LPC clock run	$\overline{\text{CLKRUN}}$	P82	Input/output ^{*1 *2}	LCLK restart request signal of serial host interrupt request
LPC power-down	$\overline{\text{LPCPD}}$	P83	Input ^{*1}	LPC module shutdown signal

- Notes: 1. Pin state monitoring input is possible in addition to the LPC interface control input/output function.
2. Only 0 can be output. If 1 is output, the pin goes to the high-impedance state. An external resistor is necessary to pull the signal up to V_{CC} .

- Host interface control register 1 (HICR1)
- Host interface control register 2 (HICR2)
- Host interface control register 3 (HICR3)
- LPC channel 3 address registers (LADR3H, LADR3L)
- Input data register 1 (IDR1)
- Output data register 1 (ODR1)
- Status register 1 (STR1)
- Input data register 2 (IDR2)
- Output data register 2 (ODR2)
- Status register 2 (STR2)
- Input data register 3 (IDR3)
- Output data register 3 (ODR3)
- Status register 3 (STR3)
- Bidirectional data registers 0 to 15 (TWR0 to TWR15)
- SERIRQ control register 0 (SIRQCR0)
- SERIRQ control register 1 (SIRQCR1)
- Host interface select register (HISEL)

Bit	Bit Name	Initial Value	Slave	Host	Description
7	LPC3E	0	R/W	—	LPC Enable 3 to 1
6	LPC2E	0	R/W	—	Enable or disable the host interface func single-chip mode. When the host interfac (one of the three bits is set to 1), process transfer between the slave processor (th the host processor is performed using pi LAD0, LFRAME, LRESET, LCLK, SERIF CLKRUN, and LPCPD.
5	LPC1E	0	R/W	—	<ul style="list-style-type: none"> • LPC3E 0: LPC channel 3 operation is disabled No address (LADR3) matches for IDR3, STR3, or TWR0 to TWR15 1: LPC channel 3 operation is enabled • LPC2E 0: LPC channel 2 operation is disabled No address (H'0062, 66) matches for ID STR2 1: LPC channel 2 operation is enabled • LPC1E 0: LPC channel 1 operation is disabled No address (H'0060, 64) matches for ID STR1 1: LPC channel 1 operation is enabled

bit for P81 must not be set to 1.

0: Fast A20 gate function disabled

- Other function of pin P81 is enabled
- GA20 output internal state is initialized

1: Fast A20 gate function enabled

- GA20 pin output is open-drain (external pull-up resistor required)

3	SDWNE	0	R/W	—	LPC Software Shutdown Enable
---	-------	---	-----	---	------------------------------

Controls host interface shutdown. For details of the host interface shutdown function, and the scope of the host interface shutdown function, see Section 19.4.4, Host Interface Shutdown Function.

0: Normal state, LPC software shutdown state is enabled

[Clearing conditions]

- Writing 0
- LPC hardware reset or LPC software reset
- LPC hardware shutdown release (rising edge of $\overline{\text{LPCPD}}$ signal)

1: LPC hardware shutdown state setting enabled

- Hardware shutdown state when $\overline{\text{LPCPD}}$ is low

[Setting condition]

- Writing 1 after reading SDWNE = 0

for P80 must not be set to 1.

PMEE PMEB

0 x: PME output disabled, other pin is enabled

1 0: PME output enabled, $\overline{\text{PME}}$ goes to 0 level

1 1: PME output enabled, $\overline{\text{PME}}$ high-impedance

1 LSMIE 0 R/W —

LSMI output Enable

Controls LSMI output in combination with bit in HICR1. LSMI pin output is open-drain. An external pull-up resistor is needed to pull up to V_{CC} .

When the LSMI output function is used, the bit for PB0 must not be set to 1.

LSMIE LSMIB

0 x: LSMI output disabled, other pin is enabled

1 0: LSMI output enabled, $\overline{\text{LSMI}}$ goes to 0 level

1 1: LSMI output enabled, $\overline{\text{LSMI}}$ high-impedance

for PB1 must not be set to 1.

LSCIE LSCIB

0 x: LSCI output disabled, other f
pin is enabled

1 0: LSCI output enabled, LSCI p
goes to 0 level

1 1: LSCI output enabled, LSCI p
high-impedance

Legend:

X: Don't care

- Bus idle, or transfer cycle not subject to processing is in progress
- Cycle type or address indeterminate transfer cycle

[Clearing conditions]

- LPC hardware reset or LPC software reset
- LPC hardware shutdown or LPC software shutdown
- Forced termination (abort) of transfer cycle subject to processing
- Normal termination of transfer cycle subject to processing

1: Host interface is performing transfer cycle processing

[Setting condition]

Match of cycle type and address

- LPC hardware reset or LPC software shutdown
- LPC hardware shutdown or LPC software shutdown
- SERIRQ is set to continuous mode
- There are no further interrupts for transfer host in quiet mode

1: LCLK restart request issued

[Setting condition]

In quiet mode, SERIRQ interrupt output becomes necessary while LCLK is stopped

5 IRQBSY 0 R —

SERIRQ Busy

Indicates that the host interface's SERIRQ is engaged in transfer processing.

0: SERIRQ transfer frame wait state

[Clearing conditions]

- LPC hardware reset or LPC software shutdown
- LPC hardware shutdown or LPC software shutdown
- End of SERIRQ transfer frame

1: SERIRQ transfer processing in progress

[Setting condition]

Start of SERIRQ transfer frame

- Writing 0
 - LPC hardware reset
- 1: LPC software reset state

[Setting condition]

Writing 1 after reading LRSTB = 0

3	SDWNB	0	R/W	—	<p>LPC Software Shutdown Bit</p> <p>Controls host interface shutdown. For details on the LPC shutdown function, and the scope of the function during initialization by an LPC reset and an LPC software reset, see section 19.4.4, Host Interface Shutdown Function (LPCPD).</p> <p>0: Normal state</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 • LPC hardware reset or LPC software reset • LPC hardware shutdown • LPC hardware shutdown release (rising edge of $\overline{\text{LPCPD}}$ signal when S₁ is asserted) <p>1: LPC software shutdown state</p> <p>[Setting condition]</p> <p>Writing 1 after reading SDWNB = 0</p>
2	PMEB	0	R/W	—	<p>PME Output Bit</p> <p>Controls PME output in combination with the $\overline{\text{LPCPD}}$ bit. For details, refer to description on the bit in HICR0.</p>

19.3.2 Host Interface Control Registers 2 and 3 (HICR2, HICR3)

Bits 6 to 0 in HICR2 control interrupts from the host interface (LPC) module to the slave processor (this LSI). Bit 7 in HICR2 and HICR3 monitor host interface pin states.

The pin states can be monitored regardless of the host interface operating state or the operating state of the functions that use pin multiplexing.

- HICR2

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	GA20	Undefined	R	—	GA20 Pin Monitor
6	LRST	0	R/(W)*	—	LPC Reset Interrupt Flag This bit is a flag that generates an ERRRI in when an LPC hardware reset occurs. 0: [Clearing condition] Writing 0 after reading LRST = 1 1: [Setting condition] LRESET pin falling edge detection

					<ul style="list-style-type: none"> LPC hardware reset and LPC software reset <p>1: [Setting condition]</p> <p>LPCPD pin falling edge detection</p>
4	ABRT	0	R/(W)*	—	<p>LPC Abort Interrupt Flag</p> <p>This bit is a flag that generates an ERR1 interrupt when a forced termination (abort) of an LPC cycle occurs.</p> <p>0: [Clearing conditions]</p> <ul style="list-style-type: none"> Writing 0 after reading ABRT = 1 LPC hardware reset and LPC software reset LPC hardware shutdown and LPC software shutdown <p>1: [Setting condition]</p> <p>$\overline{\text{LFRAME}}$ pin falling edge detection during transfer cycle</p>
3	IBFIE3	0	R/W	—	<p>IDR3 and TWR Receive Completion Interrupt Enable</p> <p>Enables or disables IBFI3 interrupt to the processor (this LSI).</p> <p>0: Input data register IDR3 and TWR receive completed interrupt requests disabled</p> <p>1: [When TWRIE = 0 in LADR3] Input data register (IDR3) receive completed interrupt requests enabled [When TWRIE = 1 in LADR3] Input data register (IDR3) and TWR receive completed interrupt requests enabled</p>

1	IBFIE1	0	R/W	—	interrupt requests enabled IDR1 Receive Completion Interrupt Enable Enables or disables IBF11 interrupt to the processor (this LSI). 0: Input data register (IDR1) receive completion interrupt requests disabled 1: Input data register (IDR1) receive completion interrupt requests enabled
0	ERRIE	0	R/W	—	Error Interrupt Enable Enables or disables ERRI interrupt to the processor (this LSI). 0: Error interrupt requests disabled 1: Error interrupt requests enabled

Note: * Only 0 can be written to bits 6 to 4, to clear the flag.

- HICR3

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	LFRAME	Undefined	R	—	$\overline{\text{LFRAME}}$ Pin Monitor
6	CLKRUN	Undefined	R	—	$\overline{\text{CLKRUN}}$ Pin Monitor
5	SERIRQ	Undefined	R	—	$\overline{\text{SERIRQ}}$ Pin Monitor
4	LRESET	Undefined	R	—	$\overline{\text{LRESET}}$ Pin Monitor
3	LPCPD	Undefined	R	—	$\overline{\text{LPCPD}}$ Pin Monitor
2	PME	Undefined	R	—	$\overline{\text{PME}}$ Pin Monitor
1	LSMI	Undefined	R	—	$\overline{\text{LSMI}}$ Pin Monitor
0	LSCI	Undefined	R	—	$\overline{\text{LSCI}}$ Pin Monitor

7	Bit 15	0	R/W	Channel 3 Address Bits 15 to 8:
6	Bit 14	0	R/W	When LPC3E = 1, an I/O address received in a cycle is compared with the contents of LADR3.
5	Bit 13	0	R/W	determining an IDR3, ODR3, or STR3 address.
4	Bit 12	0	R/W	of LADR3 is regarded as 0, and the value of bit
3	Bit 11	0	R/W	When determining a TWR0 to TWR15 address
2	Bit 10	0	R/W	of LADR3 is inverted, and the values of bits 3 to
1	Bit 9	0	R/W	ignored. Register selection according to the bit
0	Bit 8	0	R/W	address match determination is as shown in ta

- LADR3L

Bit	Bit Name	Initial Value	R/W	Description
7	Bit 7	0	R/W	Channel 3 Address Bits 7 to 3
6	Bit 6	0	R/W	
5	Bit 5	0	R/W	
4	Bit 4	0	R/W	
3	Bit 3	0	R/W	
2	—	0	R/W	Reserved This bit is readable/writable, however, only 0 should be written to this bit.
1	Bit 1	0	R/W	Channel 3 Address Bit 1
0	TWRE	0	R/W	Bidirectional Data Register Enable Enables or disables bidirectional data register operation. 0: TWR operation is disabled TWR-related I/O address match determination is disabled. 1: TWR operation is enabled

Bit 4	0	0	0	0	I/O write	TWR0MW write
Bit 4	0	0	0	1	I/O write	TWR1 to TWR15 write
	1	1	1	1		
Bit 4	0	0	0	0	I/O read	TWR0SW read
Bit 4	0	0	0	1	I/O read	TWR1 to TWR15 read
	1	1	1	1		

19.3.4 Input Data Registers 1 to 3 (IDR1 to IDR3)

The IDR registers are 8-bit read-only registers for the slave processor (this LSI), and 8-bit read-only registers for the host processor. The registers selected from the host according to the I/O address are shown in the following table. For information on IDR3 selection, see section 19.3.3. Data transferred in an LPC I/O write cycle is latched into the selected register. The state of bit 2 of the I/O address is latched into the C/D bit in the register to indicate whether the written information is a command or data. The initial values of IDR1 to IDR3 are undefined.

Bits 15 to 4	I/O Address				Transfer Cycle	Host Register Selected
	Bit 3	Bit 2	Bit 1	Bit 0		
0000 0000 0110	0	0	0	0	I/O write	IDR1 write, C/D $\bar{1}$
0000 0000 0110	0	1	0	0	I/O write	IDR1 write, C/D1
0000 0000 0110	0	0	1	0	I/O write	IDR2 write, C/D $\bar{2}$
0000 0000 0110	0	1	1	0	I/O write	IDR2 write, C/D2

Bits 15 to 4	Bit 3	Bit 2	Bit 1	Bit 0	Cycle	Host Register
0000 0000 0110	0	0	0	0	I/O read	ODR1 read
0000 0000 0110	0	0	1	0	I/O read	ODR2 read

19.3.6 Bidirectional Data Registers 0 to 15 (TWR0 to TWR15)

The TWR registers are sixteen 8-bit readable/writable registers to both the slave processor (this LSI) and the host processor. In TWR0, however, two registers (TWR0MW and TWR0SW) are allocated to the same address for both the host address and the slave address. TWR0MW is a write-only register for the host processor, and a read-only register for the slave processor. TWR0SW is a write-only register for the slave processor and a read-only register for the host processor. When the host and slave processors begin a write, after the respective TWR registers have been written to, access right arbitration for simultaneous access is performed by the status flags to see if those writes were valid. For the registers selected from the host address, see section 19.3.3, LPC Channel 3 Address Register (LADR3).

Data transferred in an LPC I/O write cycle is written to the selected register; in an LPC I/O read cycle, the data in the selected register is transferred to the host. The initial values of TWR0 to TWR15 are undefined.

19.3.7 Status Registers 1 to 3 (STR1 to STR3)

The STR registers are 8-bit registers that indicate status information during host interface processing. Bits 3, 1, and 0 of STR1 to STR3, and bits 7 to 4 of STR3, are read-only bits for the host processor and the slave processor (this LSI). However, only 0 can be written to bits 3, 1, and 0 of STR1 to STR3 and bits 6 and 4 of STR3, from the slave processor (this LSI), in order to clear the status flags to 0. The registers selected from the host processor according to the I/O address are listed in the following table. For information on STR3 selection, see section 19.3.3, LPC Channel 3 Address Register (LADR3). In an LPC I/O read cycle, the data in the selected register is transferred to the host processor. The initial values of STR1 to STR3 are H'00.

Bit	Bit Name	Initial Value	Slave	Host	Description
7	DBU17	0	R/W	R	Defined by User
6	DBU16	0	R/W	R	The user can use these bits as necessary
5	DBU15	0	R/W	R	
4	DBU14	0	R/W	R	
3	C/D $\bar{1}$	0	R	R	Command/Data When the host processor writes to an IDR bit 2 of the I/O address is written into this bit to indicate whether IDR contains data or a command. 0: Contents of data register (IDR) are data 1: Contents of data register (IDR) are a command
2	DBU12	0	R/W	R	Defined by User The user can use this bit as necessary.
1	IBF1	0	R	R	Input Buffer Full Set to 1 when the host processor writes to the slave processor. This bit is an internal interrupt source to the slave processor (this LSI). IBF is cleared to 0 when the slave processor reads IDR. The IBF1 flag setting and clearing conditions are different when the fast A20 gate is used. For details, see table 19.3. 0: [Clearing condition] When the slave processor reads IDR 1: [Setting condition] When the host processor writes to IDR during a write cycle

cycle, or the slave processor writes 0 to
 1: [Setting condition]
 When the slave processor writes to ODR

Note: * Only 0 can be written to clear the flag.

- STR2

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	DBU27	0	R/W	R	Defined by User
6	DBU26	0	R/W	R	The user can use these bits as necessary.
5	DBU25	0	R/W	R	
4	DBU24	0	R/W	R	
3	$\overline{C/D}2$	0	R	R	Command/Data When the host processor writes to an ID bit 2 of the I/O address is written into this indicate whether IDR contains data or a 0: Contents of data register (IDR) are da 1: Contents of data register (IDR) are a c
2	DBU22	0	R/W	R	Defined by User The user can use this bit as necessary.

different when the fast A20 gate is used. For more information, see table 19.3.

0: [Clearing condition]

When the slave processor reads IDR

1: [Setting condition]

When the host processor writes to IDR using the next cycle

0	OBF2	0	R/(W)*	R	Output Buffer Full
---	------	---	--------	---	--------------------

Set to 1 when the slave processor (this LSI) writes to ODR. Cleared to 0 when the host processor reads ODR.

0: [Clearing condition]

When the host processor reads ODR using the next cycle, or the slave processor writes 0 to ODR.

1: [Setting condition]

When the slave processor writes to ODR

Note: * Only 0 can be written to clear the flag.

slave processor reads TWR15.

0: [Clearing condition]

When the slave processor reads TWR15

1: [Setting condition]

When the host processor writes to TWR
write cycle

6	OBF3B	0	R/(W)*	R	<p>Bidirectional Data Register Output Buffer</p> <p>Set to 1 when the slave processor (this LSI) reads TWR15. OBF3B is cleared to 0 when the slave processor reads TWR15.</p> <p>0: [Clearing condition]</p> <p>When the host processor reads TWR15 read cycle, or the slave processor writes to TWR15 read cycle, OBF3B bit</p> <p>1: [Setting condition]</p> <p>When the slave processor writes to TWR15</p>
5	MWMF	0	R	R	<p>Master Write Mode Flag</p> <p>Set to 1 when the host processor writes to TWR15. MWMF is cleared to 0 when the slave processor (this LSI) reads TWR15.</p> <p>0: [Clearing condition]</p> <p>When the slave processor reads TWR15</p> <p>1: [Setting condition]</p> <p>When the host processor writes to TWR15 read cycle while SWMF = 0</p>

When the host processor reads TWR15 u
 read cycle, or the slave processor writes C
 SWMF bit

1: [Setting condition]

When the slave processor writes to TWR
 MWMF = 0

3	C/D3	0	R	R	Command/Data When the host processor writes to an IDR bit 2 of the I/O address is written into this indicate whether IDR contains data or a c 0: Contents of data register (IDR) are data 1: Contents of data register (IDR) are a c
2	DBU32	0	R/W	R	Defined by User The user can use this bit as necessary.
1	IBF3A	0	R	R	Input Buffer Full Set to 1 when the host processor writes to bit is an internal interrupt source to the sla processor (this LSI). IBF is cleared to 0 w slave processor reads IDR. The IBF1 flag setting and clearing conditio different when the fast A20 gate is used. F see table 19.3. 0: [Clearing condition] When the slave processor reads IDR 1: [Setting condition] When the host processor writes to IDR us write cycle

cycle, or the slave processor writes 0 to
 1: [Setting condition]
 When the slave processor writes to ODR

Note: * Only 0 can be written to clear the flag.

- STR3 (TWRE = 0 and SELSTR3 = 1)

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	DBU37	0	R/W	R	Defined by User
6	DBU36	0	R/W	R	The user can use these bits as necessary.
5	DBU35	0	R/W	R	
4	DBU34	0	R/W	R	
3	C/D $\bar{3}$	0	R	R	Command/Data When the host processor writes to an ID bit 2 of the I/O address is written into this indicate whether IDR contains data or a 0: Contents of data register (IDR) are da 1: Contents of data register (IDR) are a c
2	DBU32	0	R/W	R	Defined by User The user can use this bit as necessary.

different when the fast A20 gate is used. For more information, see table 19.3.

0: [Clearing condition]

When the slave processor reads IDR

1: [Setting condition]

When the host processor writes to IDR using the next write cycle

0	OBF3A	0	R/(W)*	R	Output Buffer Full
---	-------	---	--------	---	--------------------

Set to 1 when the slave processor (this L5) reads ODR. OBF3A is cleared to 0 when the host processor reads ODR.

0: [Clearing condition]

When the host processor reads ODR using the next read cycle, or the slave processor writes 0 to the ODR register during the next write cycle.

1: [Setting condition]

When the slave processor writes to ODR

Note: * Only 0 can be written to clear the flag.

19.3.8 SERIRQ Control Registers 0 and 1 (SIRQCR0, SIRQCR1)

The SIRQCR registers contain status bits that indicate the SERIRQ operating mode and specify SERIRQ interrupt sources.

[Clearing conditions]

- LPC hardware reset, LPC software reset
- Specification by SERIRQ transfer cycle
frame

1: Quiet mode

[Setting condition]

Specification by SERIRQ transfer cycle s

6	SELREQ	0	R/W	—	Start Frame Initiation Request Select
---	--------	---	-----	---	---------------------------------------

Selects whether start frame initiation is requested when one or more interrupt requests are cleared, in quiet mode.

0: Start frame initiation is requested when interrupt requests are cleared in quiet mode.

1: Start frame initiation is requested when more interrupt requests are cleared in quiet mode.

5	IEDIR	0	R/W	—	Interrupt Enable Direct Mode
---	-------	---	-----	---	------------------------------

Specifies whether LPC channel 2 and channel 3 SERIRQ interrupt source (SMI, IRQ6, IRQ7, IRQ10, IRQ11) generation is conditional upon OBF. When OBF is controlled only by the host interrupt enable bit.

0: Host interrupt is requested when host interrupt enable bit and corresponding OBF are both 1.

1: Host interrupt is requested when host interrupt enable bit is set to 1.

- Writing 0 to SMIE3B
 - LPC hardware reset, LPC software reset
 - Clearing OBF3B to 0 (when IEDIR = 0)
- 1: [When IEDIR = 0]
 Host SMI interrupt request by setting OBF3B to 1 is enabled
 [When IEDIR = 1]
 Host SMI interrupt is requested
 [Setting condition]
 Writing 1 after reading SMIE3B = 0

3	SMIE3A	0	R/W	—	<p>Host SMI Interrupt Enable 3A</p> <p>Enables or disables a host SMI interrupt request by OBF3A when OBF3A is set by an ODR3 write.</p> <p>0: Host SMI interrupt request by OBF3A and SMIE3A is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to SMIE3A • LPC hardware reset, LPC software reset • Clearing OBF3A to 0 (when IEDIR = 0) <p>1: [When IEDIR = 0] Host SMI interrupt request by setting OBF3A to 1 is enabled [When IEDIR = 1] Host SMI interrupt is requested [Setting condition] Writing 1 after reading SMIE3A = 0</p>
---	--------	---	-----	---	--

- Writing 0 to SMIE2
 - LPC hardware reset, LPC software re
 - Clearing OBF2 to 0 (when IEDIR = 0
- 1: [When IEDIR = 0]
 Host SMI interrupt request by setting
 is enabled
 [When IEDIR = 1]
 Host SMI interrupt is requested
 [Setting condition]
 Writing 1 after reading SMIE2 = 0

1	IRQ12E1	0	R/W	—	<p>Host IRQ12 Interrupt Enable 1</p> <p>Enables or disables a host IRQ12 interrupt when OBF1 is set by an ODR1 write.</p> <p>0: Host IRQ12 interrupt request by OBF1 IRQ12E1 is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to IRQ12E1 • LPC hardware reset, LPC software re • Clearing OBF1 to 0 <p>1: Host IRQ12 interrupt request by setting is enabled</p> <p>[Setting condition] Writing 1 after reading IRQ12E1 = 0</p>
---	---------	---	-----	---	--

- Writing 0 to IRQ1E1
 - LPC hardware reset, LPC software reset
 - Clearing OBF1 to 0
- 1: Host IRQ1 interrupt request by setting OBF1 to 1 is enabled
- [Setting condition]
- Writing 1 after reading IRQ1E1 = 0
-

- SIRQCR1

Bit	Bit Name	Initial Value	R/W		Description
			Slave	Host	
7	IRQ11E3	0	R/W	—	<p>Host IRQ11 Interrupt Enable 3</p> <p>Enables or disables a host IRQ11 interrupt request by setting OBF3A to 1 when OBF3A is set by an ODR3 write.</p> <p>0: Host IRQ11 interrupt request by OBF3A to 1. IRQ11E3 is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to IRQ11E3 • LPC hardware reset, LPC software reset • Clearing OBF3A to 0 (when IEDIR = 0) <p>1: [When IEDIR = 0]</p> <p>Host IRQ11 interrupt request by setting OBF3A to 1 is enabled</p> <p>[When IEDIR = 1]</p> <p>Host IRQ11 interrupt is requested.</p> <p>[Setting condition]</p> <p>Writing 1 after reading IRQ11E3 = 0</p>

- Writing 0 to IRQ10E3
- LPC hardware reset, LPC software reset
- Clearing OB3FA to 0 (when IEDIR = 0)

1: [When IEDIR = 0]
 Host IRQ10 interrupt request by setting OB3FA to 1 is enabled
 [When IEDIR = 1]
 Host IRQ10 interrupt is requested.
 [Setting condition]
 Writing 1 after reading IRQ10E3 = 0

5	IRQ9E3	0	R/W	—	<p>Host IRQ9 Interrupt Enable 3</p> <p>Enables or disables a host IRQ9 interrupt request by setting OBF3A to 1 when OBF3A is set by an ODR3 write.</p> <p>0: Host IRQ9 interrupt request by OBF3A to 1 IRQ9E3 is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to IRQ9E3 • LPC hardware reset, LPC software reset • Clearing OBF3A to 0 (when IEDIR = 0) <p>1: [When IEDIR = 0] Host IRQ9 interrupt request by setting OBF3A to 1 is enabled [When IEDIR = 1] Host IRQ9 interrupt is requested. [Setting condition] Writing 1 after reading IRQ9E3 = 0</p>
---	--------	---	-----	---	--

- Writing 0 to IRQ6E3
 - LPC hardware reset, LPC software reset
 - Clearing OBF3A to 0 (when IEDIR = 0)
- 1: [When IEDIR = 0]
 Host IRQ6 interrupt request by setting
 1 is enabled
 [When IEDIR = 1]
 Host IRQ6 interrupt is requested.
 [Setting condition]
 Writing 1 after reading IRQ6E3 = 0

3	IRQ11E2	0	R/W	—	<p>Host IRQ11 Interrupt Enable 2</p> <p>Enables or disables a host IRQ11 interrupt request when OBF2 is set by an ODR2 write.</p> <p>0: Host IRQ11 interrupt request by OBF2 IRQ11E2 is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to IRQ11E2 • LPC hardware reset, LPC software reset • Clearing OBF2 to 0 (when IEDIR = 0) <p>1: [When IEDIR = 0] Host IRQ11 interrupt request by setting 1 is enabled [When IEDIR = 1] Host IRQ11 interrupt is requested. [Setting condition] Writing 1 after reading IRQ11E2 = 0</p>
---	---------	---	-----	---	--

- Writing 0 to IRQ10E2
 - LPC hardware reset, LPC software reset
 - Clearing OBF2 to 0 (when IEDIR = 0)
- 1: [When IEDIR = 0]
 Host IRQ10 interrupt request by setting OBF2 = 1 is enabled
 [When IEDIR = 1]
 Host IRQ10 interrupt is requested.
 [Setting condition]
 Writing 1 after reading IRQ10E2 = 0

1	IRQ9E2	0	R/W	—	<p>Host IRQ9 Interrupt Enable 2</p> <p>Enables or disables a host IRQ9 interrupt request by setting OBF2 when OBF2 is set by an ODR2 write.</p> <p>0: Host IRQ9 interrupt request by OBF2 is disabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Writing 0 to IRQ9E2 • LPC hardware reset, LPC software reset • Clearing OBF2 to 0 (when IEDIR = 0) <p>1: [When IEDIR = 0] Host IRQ9 interrupt request by setting OBF2 = 1 is enabled [When IEDIR = 1] Host IRQ9 interrupt is requested. [Setting condition] Writing 1 after reading IRQ9E2 = 0</p>
---	--------	---	-----	---	--

- Writing 0 to IRQ6E2
- LPC hardware reset, LPC software re...
- Clearing OBF2 to 0 (when IEDIR = 0)

1: [When IEDIR = 0]

Host IRQ6 interrupt request by setting OB...
enabled

[When IEDIR = 1]

Host IRQ6 interrupt is requested.

[Setting condition]

Writing 1 after reading IRQ6E2 = 0

Selects the function of bits 7 to 4 in STR3 in combination with the TWRE bit in LADDR. For a detailed description on STR3 in section 19.3.7, See Registers 1 to 3 (STR1 to STR3), for details.

0: Bits 7 to 4 in STR3 are status bits of the host interface.

1: [When TWRE = 1]

Bits 7 to 4 in STR3 are status bits of the host interface.

[When TWRE = 0]

Bits 7 to 4 in STR3 are user bits.

6	SELIRQ11 0	W	—	SERIRQ Output Select
5	SELIRQ10 0	W	—	Selects the pin output status of host interrupt requests (HIRQ11, HIRQ10, HIRQ9, HIRQ8, HIRQ7, HIRQ6, HIRQ5, HIRQ4, HIRQ3, HIRQ2, HIRQ1, and HIRQ0) of the LPC.
4	SELIRQ9 0	W	—	
3	SELIRQ6 0	W	—	
2	SELSMI 0	W	—	0: [When host interrupt request is cleared] SERIRQ pin output is in the high-impedance state.
1	SELIRQ12 1	W	—	
0	SELIRQ1 1	W	—	[When host interrupt request is set] SERIRQ pin output is 0. 1: [When host interrupt request is cleared] SERIRQ pin output is 0. [When host interrupt request is set] SERIRQ pin output is in the high-impedance state.



host interface's input/output pins.

Use the following procedure to activate the host interface after a reset release.

1. Read the signal line status and confirm that the LPC module can be connected. Also confirm that the LPC module is initialized internally.
2. When using channel 3, set LADR3 to determine the channel 3 I/O address and whether unidirectional or bidirectional data registers are to be used.
3. Set the enable bit (LPC3E to LPC1E) for the channel to be used.
4. Set the enable bits (GA20E, PMEE, LSMIE, and LSCIE) for the additional functions to be used.
5. Set the selection bits for other functions (SDWNE, IEDIR).
6. As a precaution, clear the interrupt flags (LRST, SDWN, ABRT, OBF). Read IDR to clear IBF.
7. Set interrupt enable bits (IBFIE3 to IBFIE1, ERRIE) as necessary.

19.4.2 LPC I/O Cycles

There are ten kinds of LPC transfer cycle: memory read, memory write, I/O read, I/O write, DMA read, DMA write, bus master memory read, bus master memory write, bus master I/O read, bus master I/O write. Of these, the chip's LPC supports only I/O read and I/O write cycles.

An LPC transfer cycle is started when the $\overline{\text{LFRAME}}$ signal goes low in the bus idle state. When the $\overline{\text{LFRAME}}$ signal goes low when the bus is not idle, this means that a forced termination of the LPC transfer cycle has been requested.

In an I/O read cycle or I/O write cycle, transfer is carried out using LAD3 to LAD0 in the following order, in synchronization with LCLK. The host can be made to wait by sending a value other than B'0000 in the slave's synchronization return cycle, but with the chip's value of B'0000 is always returned.

1	Start	Host	0000	Start	Host
2	Cycle type/direction	Host	0000	Cycle type/direction	Host
3	Address 1	Host	Bits 15 to 12	Address 1	Host
4	Address 2	Host	Bits 11 to 8	Address 2	Host
5	Address 3	Host	Bits 7 to 4	Address 3	Host
6	Address 4	Host	Bits 3 to 0	Address 4	Host
7	Turnaround (recovery)	Host	1111	Data 1	Host
8	Turnaround	None	ZZZZ	Data 2	Host
9	Synchronization	Slave	0000	Turnaround (recovery)	Host
10	Data 1	Slave	Bits 3 to 0	Turnaround	None
11	Data 2	Slave	Bits 7 to 4	Synchronization	Slave
12	Turnaround (recovery)	Slave	1111	Turnaround (recovery)	Slave
13	Turnaround	None	ZZZZ	Turnaround	None

The timing of the $\overline{\text{LFRAME}}$, LCLK, and LAD signals is shown in figures 19.2 and 19.3.

			direction, and size						
Number of clocks	1	1	4	2	1	2	2	2	1

Figure 19.2 Typical LFRAME Timing

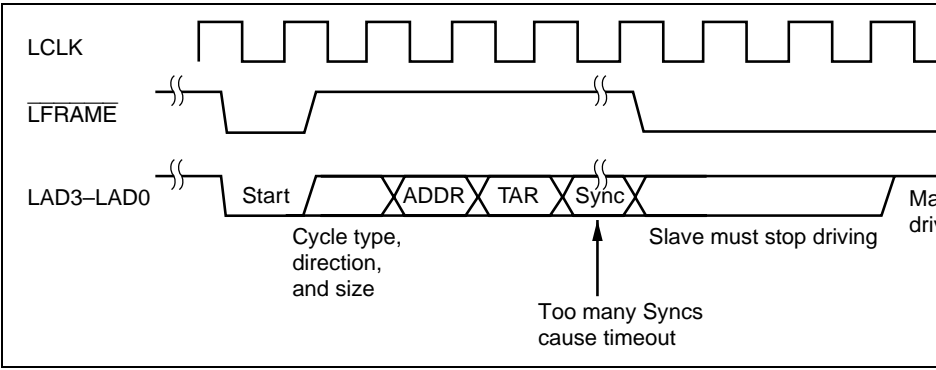


Figure 19.3 Abort Mechanism

Regular A20 Gate Operation: Output of the A20 gate signal can be controlled by an H'D1 command followed by data. When the slave processor (this LSI) receives data, it normally generates an interrupt routine activated by the IBF1 interrupt to read IDR1. At this time, firmware outputs data following an H'D1 command and outputs it at the gate A20 pin.

Fast A20 Gate Operation: The internal state of GA20 output is initialized to 1 when the FGA20E bit is set to 1, P81/GA20 is used for output of a fast A20 gate signal. The state of the P81/GA20 pin can be monitored by reading the GA20 bit in HICR2.

The initial output from this pin will be a logic 1, which is the initial value. Afterward, the slave processor can manipulate the output from this pin by sending commands and data. This data is only available via the IDR1 register. The host interface decodes commands input from the slave processor. When an H'D1 host command is detected, bit 1 of the data following the host command is output from the GA20 output pin. This operation does not depend on firmware or interrupts, but is faster than the regular processing using interrupts. Table 19.3 shows the conditions that set and clear the GA20 (P81). Figure 19.4 shows the GA20 output in flowchart form. Table 19.4 indicates the GA20 output signal values.

Table 19.3 GA20 (P81) Set/Clear Timing

Pin Name	Setting Condition	Clearing Condition
GA20 (P81)	When bit 1 of the data that follows an H'D1 host command is 1	When bit 1 of the data that follows an H'D1 host command is 0

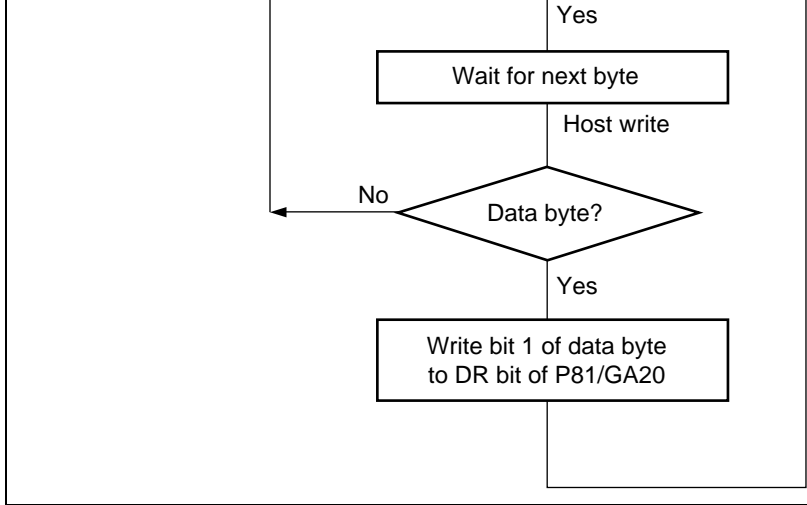


Figure 19.4 GA20 Output

1	H'D1 command	0	Q	Turn-on seq
0	0 data ^{*2}	0	0	
1	H'FF command	0	Q (0)	
1	H'D1 command	0	Q	Turn-on seq
0	1 data ^{*1}	0	1	(abbreviated)
1/0	Command other than H'FF and H'D1	1	Q (1)	
1	H'D1 command	0	Q	Turn-off seq
0	0 data ^{*2}	0	0	(abbreviated)
1/0	Command other than H'FF and H'D1	1	Q (0)	
1	H'D1 command	0	Q	Cancelled s
1	Command other than H'D1	1	Q	
1	H'D1 command	0	Q	Retriggered
1	H'D1 command	0	Q	
1	H'D1 command	0	Q	Consecutive
0	Any data	0	1/0	sequences
1	H'D1 command	0	Q (1/0)	

Notes: 1. Arbitrary data with bit 1 set to 1.
2. Arbitrary data with bit 1 cleared to 0.

Placing the slave processor in sleep mode or software standby mode is effective in reducing current dissipation in the shutdown state. If software standby mode is set, some means are provided for exiting software standby mode before clearing the shutdown state with the $\overline{\text{LPCPD}}$ signal.

If the SDWNE bit has been set to 1 beforehand, the LPC hardware shutdown state is entered at the same time as the $\overline{\text{LPCPD}}$ signal falls, and prior preparation is not possible. If the LPC software standby state is set by means of the SDWNB bit, on the other hand, the LPC software standby state cannot be cleared at the same time as the rise of the $\overline{\text{LPCPD}}$ signal. Taking these points into consideration, the following operating procedure uses a combination of LPC software standby mode and LPC hardware shutdown.

1. Clear the SDWNE bit to 0.
2. Set the ERRIE bit to 1 and wait for an interrupt by the SDWN flag.
3. When an ERRI interrupt is generated by the SDWN flag, check the host interface interrupt status flags and perform any necessary processing.
4. Set the SDWNB bit to 1 to set LPC software standby mode.
5. Set the SDWNE bit to 1 and make a transition to LPC hardware standby mode. The SDWNE bit is cleared automatically.
6. Check the state of the $\overline{\text{LPCPD}}$ signal to make sure that the $\overline{\text{LPCPD}}$ signal has not risen during steps 3 to 5. If the signal has risen, clear SDWNE to 0 to return to the state in step 1.
7. Place the slave processor in sleep mode or software standby mode as necessary.
8. If software standby mode has been set, exit software standby mode by some means independent of the LPC.
9. When a rising edge is detected in the $\overline{\text{LPCPD}}$ signal, the SDWNE bit is automatically cleared to 0. If the slave processor has been placed in sleep mode, the mode is exited by the $\overline{\text{LRESET}}$ signal input, on completion of the LPC transfer cycle, or by some other means.

$\overline{\text{LRESET}}$	P36	×	Input	Hi-Z
$\overline{\text{SERIRQ}}$	P37	O	I/O	Hi-Z
$\overline{\text{LSCI}}$	PB1	Δ	I/O	Hi-Z, only when LSCIE = 1
$\overline{\text{LSMI}}$	PB0	Δ	I/O	Hi-Z, only when LSMIE = 1
$\overline{\text{PME}}$	P80	Δ	I/O	Hi-Z, only when PMEE = 1
$\overline{\text{GA20}}$	P81	Δ	I/O	Hi-Z, only when FGA20E = 1
$\overline{\text{CLKRUN}}$	P82	O	Input	Hi-Z
$\overline{\text{LPCPD}}$	P83	×	Input	Needed to clear shutdown state

Legend:

O: Pin that is shutdown by the shutdown function

Δ: Pin that is shutdown only when the LPC function is selected by register setting

×: Pin that is not shutdown

In the LPC shutdown state, the LPC's internal state and some register bits are initialized. The order of priority of LPC shutdown and reset states is as follows.

1. System reset (reset by $\overline{\text{STBY}}$ or $\overline{\text{RES}}$ pin input, or WDT0 overflow)
 - All register bits, including bits LPC3E to LPC1E, are initialized.
2. LPC hardware reset (reset by $\overline{\text{LRESET}}$ pin input)
 - LRSTB, SDWNE, and SDWNB bits are cleared to 0.
3. LPC software reset (reset by LRSTB)
 - SDWNE and SDWNB bits are cleared to 0.
4. LPC hardware shutdown
 - SDWNB bit is cleared to 0.
5. LPC software shutdown

ERRREQ and IRQDSY flags

Host interface flags (IBF1, IBF2, IBF3A, IBF3B, MWMF, C/D $\bar{1}$ to C/D $\bar{3}$, OBF1, OBF2, OBF3A, OBF3B, SWMF, DBU), GA20 (internal state)	Initialized	Initialized	R
Host interrupt enable bits (IRQ1E1, IRQ12E1, SMIE2, IRQ6E2, IRQ9E2 to IRQ11E2, SMIE3B, SMIE3A, IRQ6E3, IRQ9E3 to IRQ11E3), Q/C flag, SELREQ bit	Initialized	Initialized	R
LRST flag	Initialized (0)	Can be set/cleared	C se
SDWN flag	Initialized (0)	Initialized (0)	C se
LRSTB bit	Initialized (0)	HR: 0 SR: 1	0
SDWNB bit	Initialized (0)	Initialized (0)	H S
SDWNE bit	Initialized (0)	Initialized (0)	H S
Host interface operation control bits (LPC3E to LPC1E, FGA20E, LADR3, IBFIE1 to IBFIE3, PMEE, PMEB, LSMIE, LSMIB, LSCIE, LSCIB, TWRE, SELSTR3, SELIRQ1, SELSMI, SELIRQ6, SELIRQ9 to SELIRQ12)	Initialized	Retained	R
$\overline{\text{LRESET}}$ signal	Input (port function)	Input	In
$\overline{\text{LPCPD}}$ signal		Input	In
LAD3 to LAD0, $\overline{\text{LFRAME}}$, LCLK, SERIRQ, $\overline{\text{CLKRUN}}$ signals		Input	H
$\overline{\text{PME}}$, $\overline{\text{LSMI}}$, LSCI, GA20 signals (when function is selected)		Output	H
$\overline{\text{PME}}$, $\overline{\text{LSMI}}$, LSCI, GA20 signals (when function is not selected)		Port function	P

Note: System reset: Reset by STBY input, RES input, or WDT overflow
LPC reset: Reset by LPC hardware reset (HR) or LPC software reset (SR)
LPC shutdown: Reset by LPC hardware shutdown (HS) or LPC software shutdo

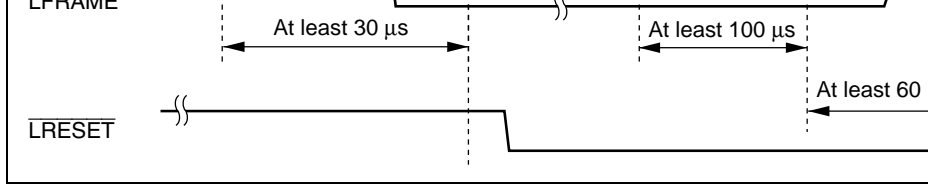


Figure 19.5 Power-Down State Termination Timing

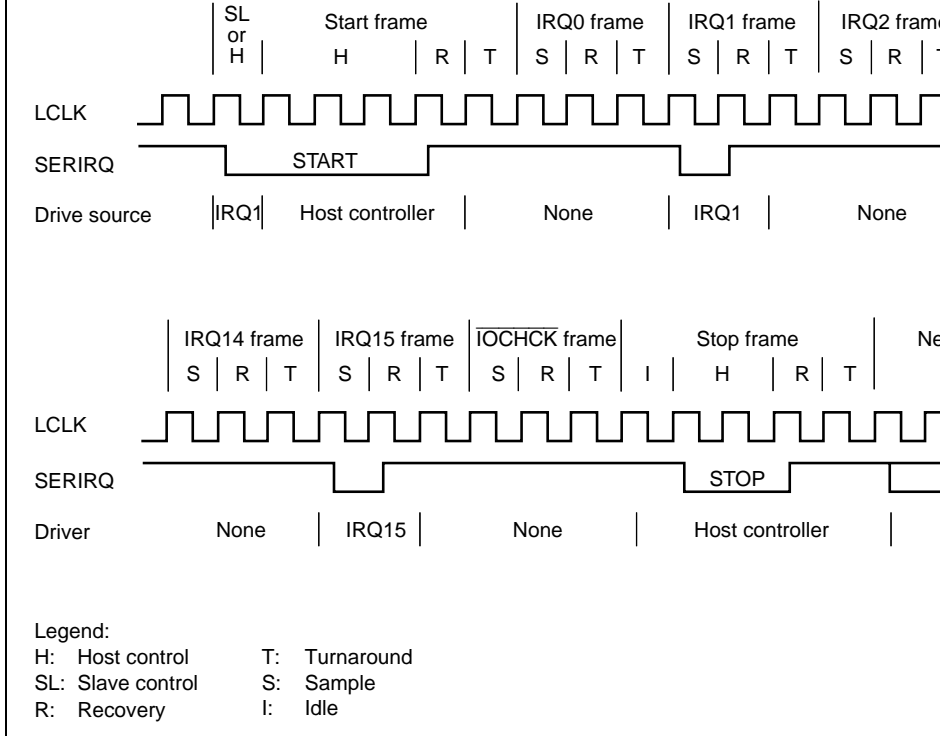


Figure 19.6 SERIRQ Timing

The serialized interrupt transfer cycle frame configuration is as follows. Two of the states comprising each frame are the recover state in which the SERIRQ signal is returned to its initial state at the end of the frame, and the turnaround state in which the SERIRQ signal is not driven. The recover state must be driven by the host or slave processor that was driving the preceding

4	IRQ3	Slave	3	
5	IRQ4	Slave	3	
6	IRQ5	Slave	3	
7	IRQ6	Slave	3	Drive possible in LPC channels 2 a
8	IRQ7	Slave	3	
9	IRQ8	Slave	3	
10	IRQ9	Slave	3	Drive possible in LPC channels 2 a
11	IRQ10	Slave	3	Drive possible in LPC channels 2 a
12	IRQ11	Slave	3	Drive possible in LPC channels 2 a
13	IRQ12	Slave	3	Drive possible in LPC channel 1
14	IRQ13	Slave	3	
15	IRQ14	Slave	3	
16	IRQ15	Slave	3	
17	IOCHCK	Slave	3	
18	Stop	Host	Undefined	First, 1 or more idle states, then 2 0-driven by host 2 states: Quiet mode next 3 states: Continuous mode next

There are two modes—continuous mode and quiet mode—for serialized interrupts. The interrupt mode selected for the next transfer cycle is selected by the stop frame of the serialized interrupt transfer cycle that ended before that cycle.

In continuous mode, the host initiates host interrupt transfer cycles at regular intervals. In quiet mode, the slave processor with interrupt sources requiring a request can also initiate a transfer cycle, in addition to the host. In quiet mode, since the host does not necessarily initiate interrupt transfer cycles, it is possible to suspend the clock (LCLK) supply and enter the down state. In order for a slave to transfer an interrupt request in this case, a request to

interrupt request is generated the CLKRUN signal is driven and a clock (LCLK) restart is sent to the host. The timing for this operation is shown in figure 19.7.

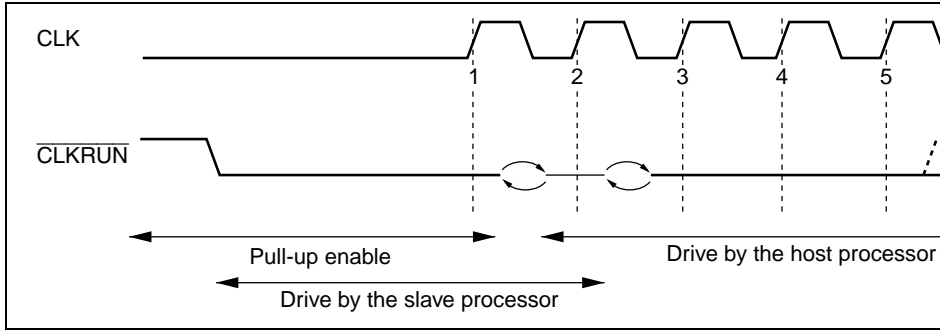


Figure 19.7 Clock Start Request Timing

Cases other than SERIRQ in quiet mode when clock restart is required must be handled different protocol, using the $\overline{\text{PME}}$ signal, etc.

corresponding enable bit.

Table 19.7 Receive Complete Interrupts and Error Interrupt

Interrupt	Description
IBF11	When IBFIE1 is set to 1 and IDR1 reception is completed
IBF12	When IBFIE2 is set to 1 and IDR2 reception is completed
IBF13	When IBFIE3 is set to 1 and IDR3 reception is completed, or when TWR15 and IBFIE3 are set to 1 and reception is completed up to TWR15
ERR1	When ERRIE is set to 1 and one of LRST, SDWN and ABRT is set to 1

19.5.2 SMI, HIRQ1, HIRQ6, HIRQ9 to HIRQ12

The host interface can request seven kinds of host interrupt by means of SERIRQ. HIRQ1 to HIRQ12 are used on LPC channel 1 only, while SMI, HIRQ6, HIRQ9 to HIRQ11 can be requested from LPC channel 2 or 3.

There are two ways of clearing a host interrupt request.

When the IEDIR bit is cleared to 0 in SIRQCR0, host interrupt sources and LPC channels are linked to the host interrupt request enable bits. When the OBF flag is cleared to 0 by a TWR15 or ODR in the corresponding LPC channel, the corresponding host interrupt request enable bit is automatically cleared to 0, and the host interrupt request is cleared.

When the IEDIR bit is set to 1 in SIRQCR0, LPC channel 2 and 3 interrupt requests are linked only upon the host interrupt enable bits. The host interrupt enable bit is not cleared when the OBF flag of channel 2 or 3 is cleared. Therefore, SMIE2, SMIE3A and SMIE3B, IRQ6E2 and IRQ6E3, IRQ9E2 and IRQ9E3, IRQ10E2 and IRQ10E3, and IRQ11E2 and IRQ11E3 lose their functional differences. In order to clear a host interrupt request, it is necessary to clear the host interrupt enable bit.

HIRQ12 (independent from IEDIR)	Internal CPU writes to ODR1, then reads 0 from bit IRQ12E1 and writes 1	Internal CPU writes 0 to bit or host reads ODR1
SMI (IEDIR = 0)	Internal CPU <ul style="list-style-type: none"> writes to ODR2, then reads 0 from bit SMIE2 and writes 1 writes to ODR3, then reads 0 from bit SMIE3A and writes 1 writes to TWR15, then reads 0 from bit SMIE3B and writes 1 	Internal CPU <ul style="list-style-type: none"> writes 0 to bit SMIE2, o reads ODR2 writes 0 to bit SMIE3A, reads ODR3 writes 0 to bit SMIE3B, reads TWR15
SMI (IEDIR = 1)	Internal CPU <ul style="list-style-type: none"> reads 0 from bit SMIE2, then writes 1 reads 0 from bit SMIE3A, then writes 1 reads 0 from bit SMIE3B, then writes 1 	Internal CPU <ul style="list-style-type: none"> writes 0 to bit SMIE2 writes 0 to bit SMIE3A writes 0 to bit SMIE3B
HIRQi (i = 6, 9 to 11) (IEDIR = 0)	Internal CPU <ul style="list-style-type: none"> writes to ODR2, then reads 0 from bit IRQiE2 and writes 1 writes to ODR3, then reads 0 from bit IRQiE3 and writes 1 	Internal CPU <ul style="list-style-type: none"> writes 0 to bit IRQiE2, o reads ODR2 CPU writes 0 to bit IRQ reads ODR3
HIRQi (i = 6, 9 to 11) (IEDIR = 1)	Internal CPU <ul style="list-style-type: none"> reads 0 from bit IRQiE2, then writes 1 reads 0 from bit IRQiE3, then writes 1 	Internal CPU <ul style="list-style-type: none"> writes 0 to bit IRQiE2 writes 0 to bit IRQiE3

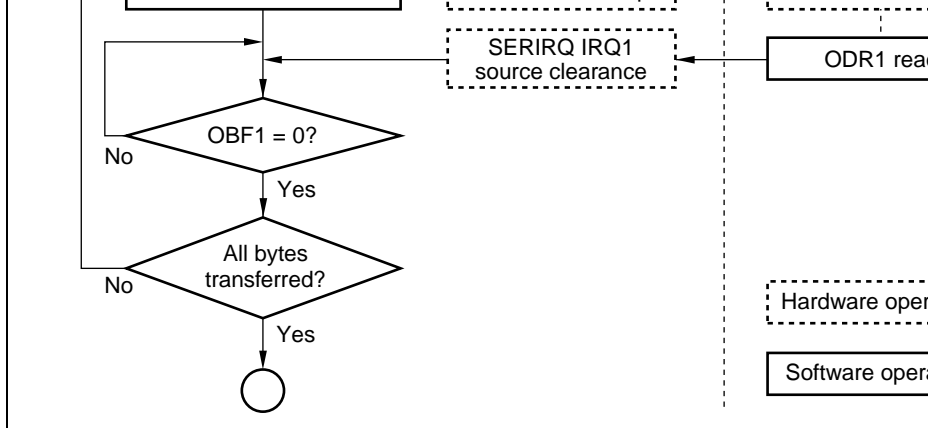


Figure 19.8 HIRQ Flowchart (Example of Channel 1)

19.6 Usage Notes

19.6.1 Module Stop Mode Setting

LPC operation can be enabled or disabled using the module stop control register. The setting is for LPC operation to be halted. Register access is enabled by canceling module stop mode. For details, refer to section 26, Power-Down Modes.

19.6.2 Notes on Using Host Interface

The host interface provides buffering of asynchronous data from the host processor and the slave processor (this LSI), but an interface protocol that uses the flags in STR must be followed to avoid data contention. For example, if the host and slave processor both try to access IDR or STR at the same time, the data will be corrupted. To prevent simultaneous accesses, IBF and OBF are used to allow access only to data for which writing has finished.

Register	Host Address when LADR3 = H'A24F	Host Address when LADR3 = H'3FD0
IDR3	H'A24A and H'A24E	H'3FD0 and H'3FD4
ODR3	H'A24A	H'3FD0
STR3	H'A24E	H'3FD4
TWR0MW	H'A250	H'3FC0
TWR0SW	H'A250	H'3FC0
TWR1	H'A251	H'3FC1
TWR2	H'A252	H'3FC2
TWR3	H'A253	H'3FC3
TWR4	H'A254	H'3FC4
TWR5	H'A255	H'3FC5
TWR6	H'A256	H'3FC6
TWR7	H'A257	H'3FC7
TWR8	H'A258	H'3FC8
TWR9	H'A259	H'3FC9
TWR10	H'A25A	H'3FCA
TWR11	H'A25B	H'3FCB
TWR12	H'A25C	H'3FCC
TWR13	H'A25D	H'3FCD
TWR14	H'A25E	H'3FCE
TWR15	H'A25F	H'3FCF

- Output voltage: 0 V to AVref
- D/A output retaining function in software standby mode

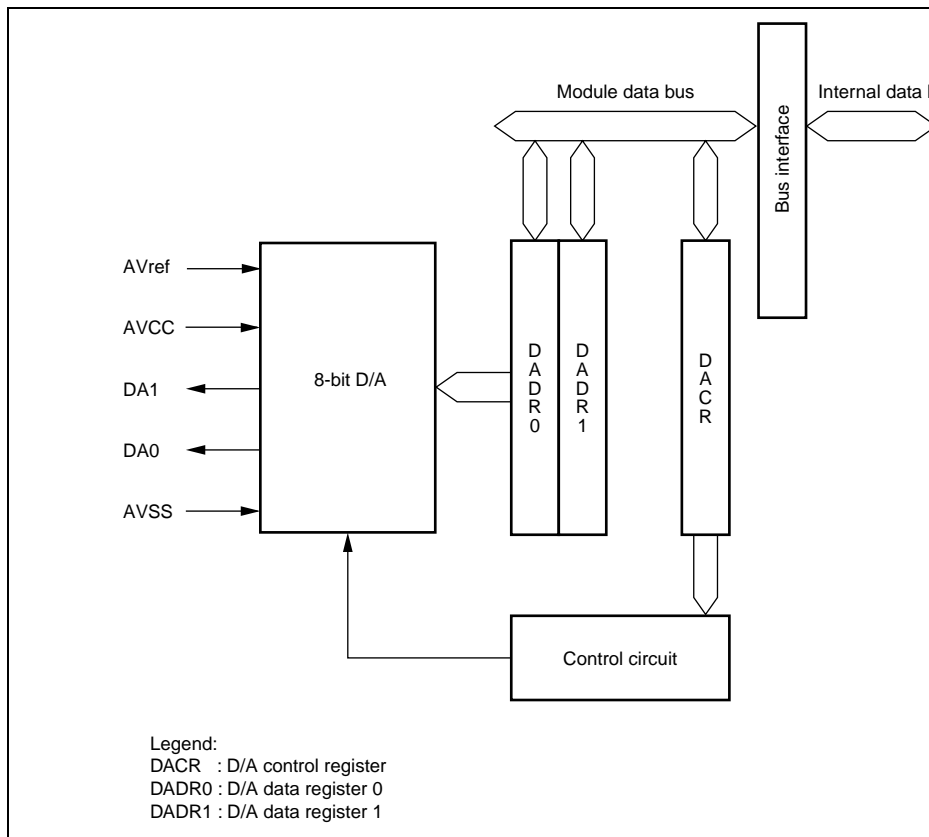


Figure 20.1 Block Diagram of D/A Converter

Analog ground pin	AVSS	Input	Analog block ground and referen
Analog output pin 0	DA0	Output	Channel 0 analog output
Analog output pin 1	DA1	Output	Channel 1 analog output
Reference power supply pin	AVref	Input	Analog block reference voltage

20.3 Register Descriptions

The D/A converter has the following registers.

- D/A data register 0 (DADR0)
- D/A data register 1 (DADR1)
- D/A control register (DACR)

20.3.1 D/A Data Registers 0 and 1 (DADR0, DADR1)

DADR0 and DADR1 are 8-bit readable/writable registers that store data for D/A conversion. When analog output is permitted, D/A data register contents are converted and output to the analog output pins. DADR0 and DADR1 are initialized to H'00.

				1: D/A conversion for channel 1 and analog output are enabled
6	DAOE0	0	R/W	D/A Output Enable 0 Controls D/A conversion and analog output. 0: Analog output DA0 is disabled 1: D/A conversion for channel 0 and analog output are enabled
5	DAE	0	R/W	D/A Enable Controls D/A conversion in conjunction with the D/AE and DAOE1 bits. When the DAE bit is cleared, D/A conversion for channels 0 and 1 is controlled by the D/AE0 and D/AE1 bits. When the DAE bit is set to 1, D/A conversion for channels 0 and 1 are controlled as one. Conversion result output is controlled by the DAOE0 and DAOE1 bits. For details, see table 20.2 below.
4 to 0	—	All 1	R	Reserved These bits are always read as 1 and cannot be written.

1	0	0	Disables D/A conversion for channel 0
			Enables D/A conversion for channel 1
		1	Enables D/A conversion for channels 0 and 1
	1	—	Enables D/A conversion for channels 0 and 1

20.4 Operation

The D/A converter incorporates two channels of the D/A circuits and can be converted individually.

When the DAOE bit in DACR is set to 1, D/A conversion is enabled and conversion results are output.

An example of D/A conversion of channel 0 is shown below. The operation timing is shown in figure 20.2.

1. Write conversion data to DADR0.
2. When the DAOE0 bit in DACR is set to 1, D/A conversion starts. After the interval of $t_{D/A}$, conversion results are output from the analog output pin DA0. The conversion results are output continuously until DADR0 is modified or the DAOE0 bit is cleared to 0. The output value is calculated by the following formula:

$$\text{DADR contents}/256 \times \text{AVref}$$

3. Conversion starts immediately after DADR0 is modified. After the interval of $t_{D/A}$, conversion results are output.
4. When the DAOE0 bit is cleared to 0, analog output is disabled.

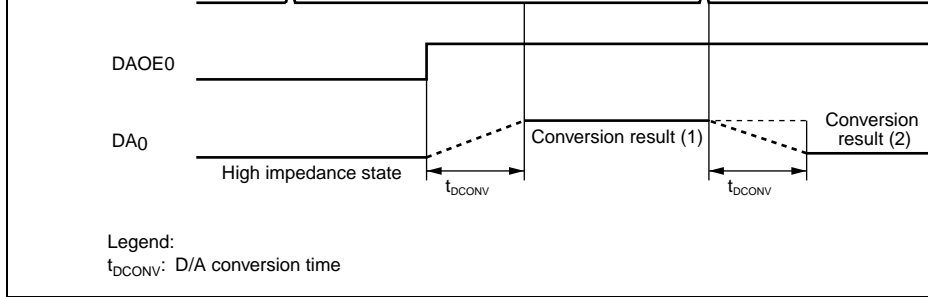


Figure 20.2 D/A Converter Operation Example

20.5 Usage Note

When this LSI enters software standby mode with D/A conversion enabled, the D/A output is retained, and the analog power supply current is equal to as during D/A conversion. If the power supply current needs to be reduced in software standby mode, clear the DAOE0 and DAE bits all to 0 to disable D/A output.

20.5.1 Module Stop Mode Setting

D/A converter operation can be enabled or disabled using the module stop control register. The initial setting is for D/A converter operation to be halted. Register access is enabled by entering module stop mode. For details, refer to section 26, Power-Down Modes.

- 10-bit resolution
- Input channels: eight analog input channels and 16 digital input channels
- Analog conversion voltage range can be specified using the reference power supply pin (AVref) as an analog reference voltage.
- Conversion time: 13.4 μ s per channel (at 10-MHz operation)
- Two kinds of operating modes
 - Single mode: Single-channel A/D conversion
 - Scan mode: Continuous A/D conversion on 1 to 4 channels
- Four data registers
 - Conversion results are held in a 16-bit data register for each channel
- Sample and hold function
- Three kinds of conversion start
 - Software, 8-bit timer (TMR) conversion start trigger, or external trigger signal.
- Interrupt request
 - A/D conversion end interrupt (ADI) request can be generated

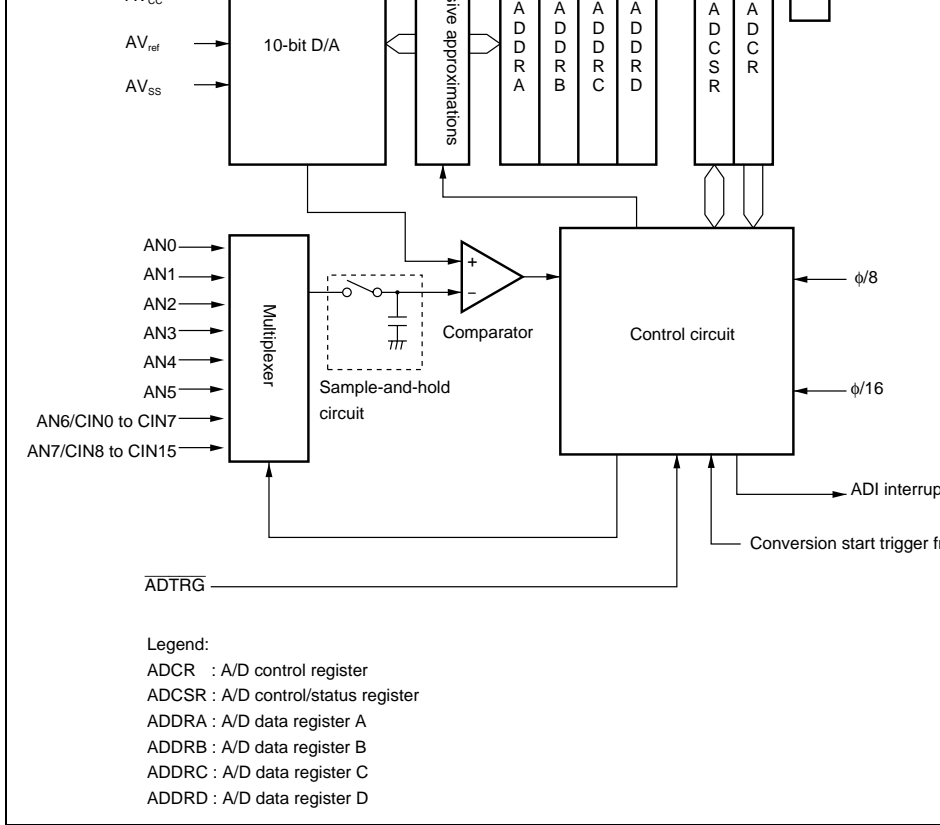


Figure 21.1 Block Diagram of A/D Converter

Table 21.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Analog power supply pin	AV_{cc}	Input	Analog block power supply and reference voltage
Analog ground pin	AV_{ss}	Input	Analog block ground and reference voltage
Reference power supply pin	AVref	Input	Reference voltage for A/D conversion
Analog input pin 0	AN0	Input	Group 0 analog input pins
Analog input pin 1	AN1	Input	
Analog input pin 2	AN2	Input	
Analog input pin 3	AN3	Input	
Analog input pin 4	AN4	Input	Group 1 analog input pins
Analog input pin 5	AN5	Input	
Analog input pin 6	AN6	Input	
Analog input pin 7	AN7	Input	
A/D external trigger input pin	\overline{ADTRG}	Input	External trigger input pin for starting A/D conversion
Expanded A/D conversion input pins 0 to 15	CIN0 to CIN15	Input	Expanded A/D conversion input (digital input channels 0 to 15) Can be used as digital input pins

- A/D control/status register (ADCSR)
- A/D control register (ADCR)
- Keyboard comparator control register (KBCOMP)

21.3.1 A/D Data Registers A to D (ADDRA to ADDR D)

There are four 16-bit read-only ADDR registers, ADDRA to ADDR D, used to store the A/D conversion. The ADDR registers, which store a conversion result for each channel in table 21.2.

The converted 10-bit data is stored to bits 15 to 6. The lower 6-bit data is always read a

The data bus between the CPU and the A/D converter is 8-bit width. The upper byte can be read directly from the CPU, but the lower byte should be read via a temporary register. The lower register contents are transferred from the ADDR when the upper byte data is read. When reading the ADDR, read the upper byte before lower byte or in word units.

Table 21.2 Analog Input Channels and Corresponding ADDR Registers

Analog Input Channel		A/D Data Register to Store A/D Conversion Results
Group 0	Group 1	
AN0	AN4	ADDRA
An1	AN5	ADDRB
AN2	AN6, or CIN0 to CIN7	ADDRC
AN3	AN7, or CIN8 to CIN15	ADDRD

- When A/D conversion ends in single r
- When A/D conversion ends on all cha specified in scan mode

[Clearing conditions]

- When 0 is written after reading ADF =
- When DTC starts by an ADI interrupt r read

6	ADIE	0	R/W	A/D Interrupt Enable Enables ADI interrupt by ADF when this b
5	ADST	0	R/W	A/D Start Setting this bit to 1 starts A/D conversion. this bit to 0 stops A/D conversion. In singl bit is cleared to 0 automatically when con the specified channel ends. In scan mode continues sequentially on the specified ch this bit is cleared to 0 by software, a reset transition to standby mode or module stop
4	SCAN	0	R/W	Scan Mode Selects the A/D conversion operating mod setting of this bit must be made when con halted (ADST = 0). 0: Single mode 1: Scan mode
3	CKS	0	R/W	Clock Select Sets A/D conversion time. The input chan must be made when conversion is halted 0: Conversion time is 266 states (max) 1: Conversion time is 134 states (max) Switch conversion time while ADST is 0.

010: AN2	010: AN0 to
011: AN3	011: AN0 to
100: AN4	100: AN4
101: AN5	101: AN4 ar
110: AN6, or CIN0 to CIN7	110: AN4 to
111: AN7, or CIN8 to CIN15	CIN0 to CIN
	111: AN4 to
	CIN0 to CIN
	or CIN8 to C

Note: * Only 0 can be written for clearing the flag.

21.3.3 A/D Control Register (ADCR)

ADCR enables A/D conversion started by an external trigger signal.

Bit	Bit Name	Initial Value	R/W	Description
7	TRGS1	0	R/W	Timer Trigger Select 1 and 0
6	TRGS0	0	R/W	Enable the start of A/D conversion by a trigger signal. Only set bits TRGS1 and TRGS0 when conversion is halted (ADST = 0). 00: A/D conversion start by external trigger 01: A/D conversion start by external trigger 10: A/D conversion start by conversion trigger TMR is enabled 11: A/D conversion start by $\overline{\text{ADTRG}}$ pin is selected
5 to 0	—	All 1	R	Reserved These bits are always read as 1 and cannot be modified.

5	IrCKS1	0	R/W																					
4	IrCKS0	0	R/W																					
3	KBADE	0	R/W	Keyboard A/D Enable (AN6, AN7) Selects whether channels 6 and 7 of the A/D converter are used as analog pins or digital pins, in combination with the KBCH2 to KBCH0 bits. For details, see the A/D converter description for bits 2 to 0. Analog pins of the A/D converter are set to digital pins (CIN0 to CIN15).																				
2	KBCH2	0	R/W	Keyboard A/D Channel Select 2 to 0																				
1	KBCH1	0	R/W	These bits select a channel of digital input pin for A/D conversion, in combination with the KBADE bit. The input channel setting must be made while the microcontroller is halted.																				
0	KBCH0	0	R/W																					
				<table border="0"> <thead> <tr> <th>Channel 6</th> <th>Channel 7</th> </tr> </thead> <tbody> <tr> <td>0xxx: Selects AN6</td> <td>AN7</td> </tr> <tr> <td>1000: Selects CIN0</td> <td>CIN8</td> </tr> <tr> <td>1001: Selects CIN1</td> <td>CIN9</td> </tr> <tr> <td>1010: Selects CIN2</td> <td>CIN10</td> </tr> <tr> <td>1011: Selects CIN3</td> <td>CIN11</td> </tr> <tr> <td>1100: Selects CIN4</td> <td>CIN12</td> </tr> <tr> <td>1101: Selects CIN5</td> <td>CIN13</td> </tr> <tr> <td>1110: Selects CIN6</td> <td>CIN14</td> </tr> <tr> <td>1111: Selects CIN7</td> <td>CIN15</td> </tr> </tbody> </table>	Channel 6	Channel 7	0xxx: Selects AN6	AN7	1000: Selects CIN0	CIN8	1001: Selects CIN1	CIN9	1010: Selects CIN2	CIN10	1011: Selects CIN3	CIN11	1100: Selects CIN4	CIN12	1101: Selects CIN5	CIN13	1110: Selects CIN6	CIN14	1111: Selects CIN7	CIN15
Channel 6	Channel 7																							
0xxx: Selects AN6	AN7																							
1000: Selects CIN0	CIN8																							
1001: Selects CIN1	CIN9																							
1010: Selects CIN2	CIN10																							
1011: Selects CIN3	CIN11																							
1100: Selects CIN4	CIN12																							
1101: Selects CIN5	CIN13																							
1110: Selects CIN6	CIN14																							
1111: Selects CIN7	CIN15																							

Legend:

x: Don't care

21.4.1 Single Mode

In single mode, A/D conversion is to be performed only once on the specified single channel. Operations are as follows.

1. A/D conversion on the specified channel is started when the ADST bit in ADCSR is set to 1 by software or an external trigger input.
2. When A/D conversion is completed, the result is transferred to the A/D data register corresponding to the channel.
3. On completion of A/D conversion, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit remains set to 1 during A/D conversion. When conversion ends, the ADST bit is automatically cleared to 0, and the A/D converter enters wait state.

21.4.2 Scan Mode

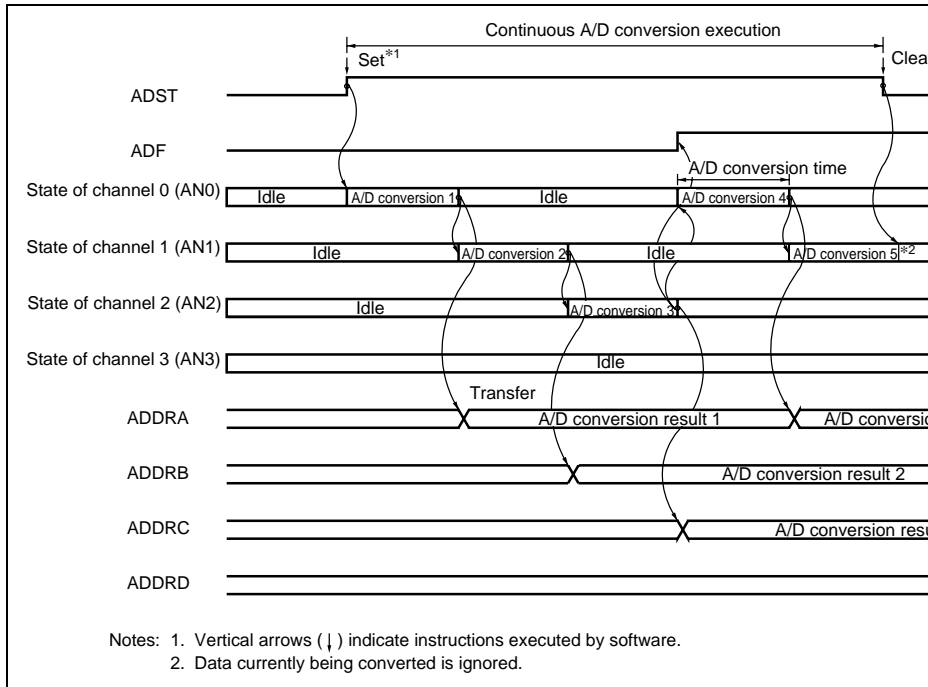
Scan mode is useful for monitoring analog inputs in a group of one or more channels. When the ADST bit is set to 1 by software, or by timer or external trigger input, A/D conversion starts on the first channel in the group (AN0 when CH2 = 0; AN4 when CH2 = 1).

When two or more channels are selected, after conversion of the first channel ends, conversion of the second channel (AN1 or AN5) starts immediately. A/D conversion continues cyclically on the selected channels until the ADST bit is cleared to 0. The conversion results are transferred to storage into the ADDR registers corresponding to the channels.

Typical operations when three channels (AN0 to AN2) are selected in scan mode are shown below.

to 1 and conversion of the first channel (AN0) starts again. If the ADIE bit is set to 1, an ADI interrupt is requested after A/D conversion ends.

- Steps 2 to 4 are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion stops. After that, if the ADST bit is set to 1, A/D conversion starts again from the first channel (AN0).



**Figure 21.2 Example of A/D Converter Operation
(Scan Mode, Channels AN0 to AN2 Selected)**

conversion time therefore varies within the ranges indicated in table 21.3.

In scan mode, the values given in table 21.3 apply to the first conversion time. In the subsequent conversions, the conversion time is 256 state (fixed) when CKS = 0 and 128 (fixed) when CKS = 1.

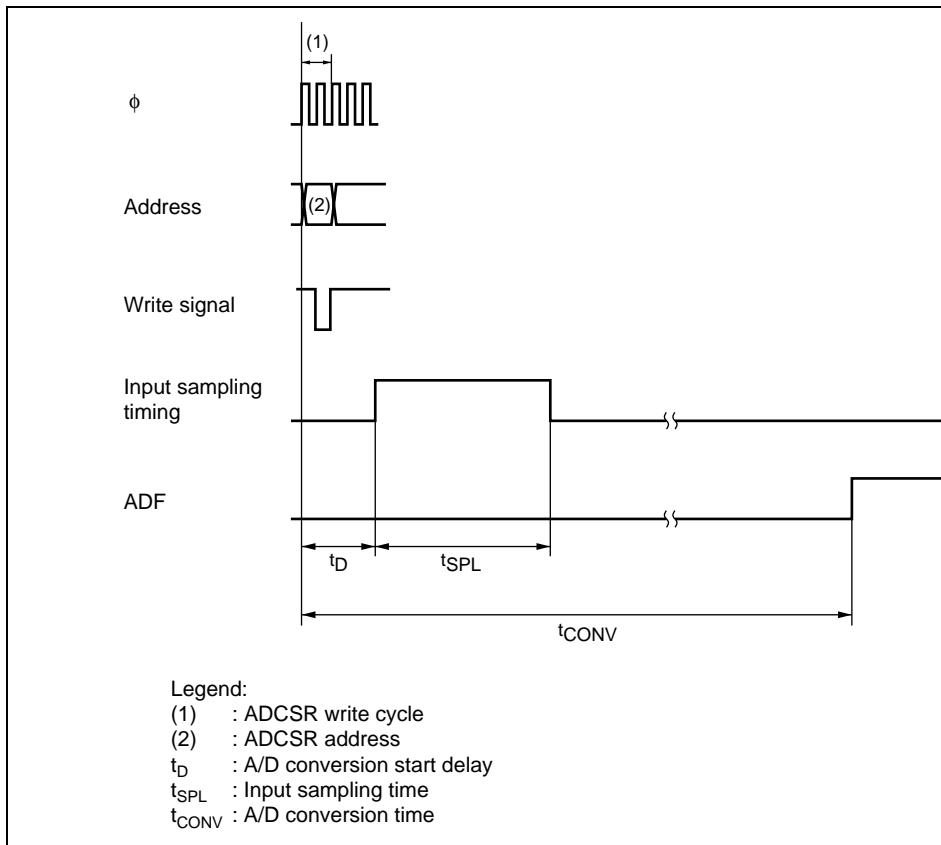


Figure 21.3 A/D Conversion Timing

21.4.4 External Trigger Input Timing

A/D conversion can be externally triggered. When the TRGS1 and TRGS0 bits are set in the ADSC register (ADCSR), external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge at the $\overline{\text{ADTRG}}$ pin sets the ADST bit to 1 in ADCSR, starting A/D conversion. Other operations, in both single and scan modes, are the same as when the ADST bit has been set to 1 by software. Figure 21.4 shows the timing.

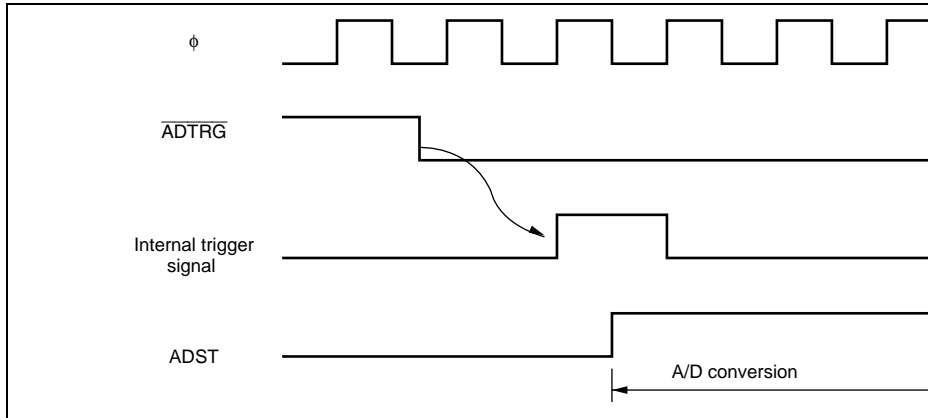


Figure 21.4 External Trigger Input Timing

This LSI's A/D conversion accuracy definitions are given below.

- Resolution
The number of A/D converter digital output codes
- Quantization error
The deviation inherent in the A/D converter, given by 1/2 LSB (see figure 21.5).
- Offset error
The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from the minimum voltage value B'0000000000 (H'000) to B'0000000001 (H'001) (see figure 21.6).
- Full-scale error
The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from B'1111111110 (H'3FE) to B'1111111111 (H'3FF) (see figure 21.6).
- Nonlinearity error
The error with respect to the ideal A/D conversion characteristics between the zero and the full-scale voltage. Does not include the offset error, full-scale error, or quantization error (see figure 21.6).
- Absolute accuracy
The deviation between the digital value and the analog input value. Includes the offset error, full-scale error, quantization error, and nonlinearity error.

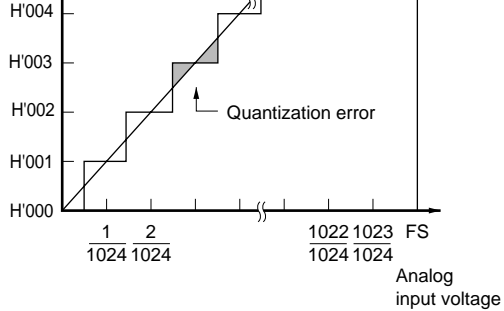


Figure 21.5 A/D Conversion Accuracy Definitions

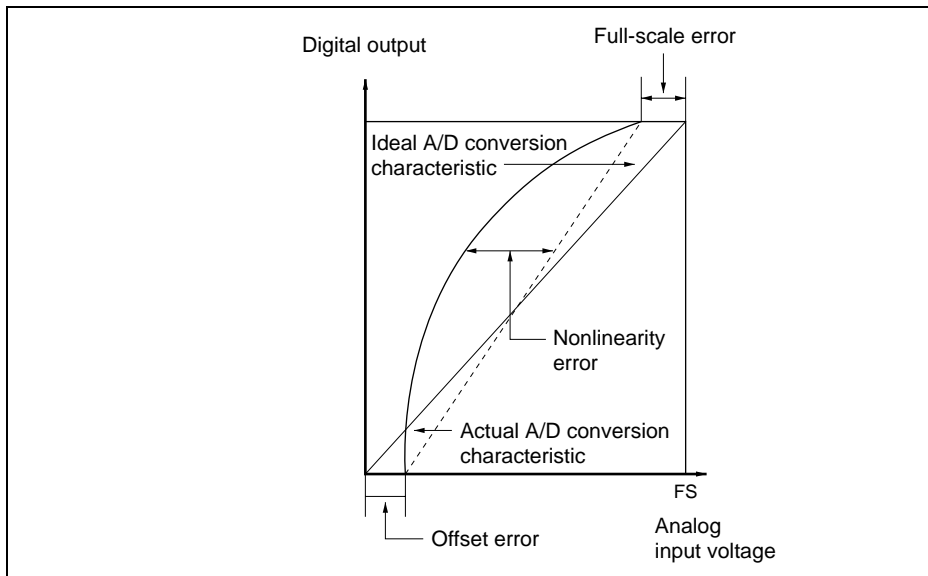


Figure 21.6 A/D Conversion Accuracy Definitions

insufficient and it may not be possible to guarantee the A/D conversion accuracy. How large capacitance is provided externally in single mode, the input load will essentially only the internal input resistance of 10 kΩ, and the signal source impedance is ignored. since a low-pass filter effect is obtained in this case, it may not be possible to follow an signal with a large differential coefficient (e.g., voltage fluctuation ratio of 5 mV/μs or (see figure 21.7). When converting a high-speed analog signal or converting in scan mode impedance buffer should be inserted. For details on the 5-V version, refer to section 28 Characteristics.

21.7.2 Influences on Absolute Accuracy

Adding capacitance results in coupling with ground, and therefore noise in ground may affect the absolute accuracy. Be sure to make the connection to an electrically stable ground as AVss.

Care is also required to insure that filter circuits do not communicate with digital signals on the mounting board, so acting as antennas.

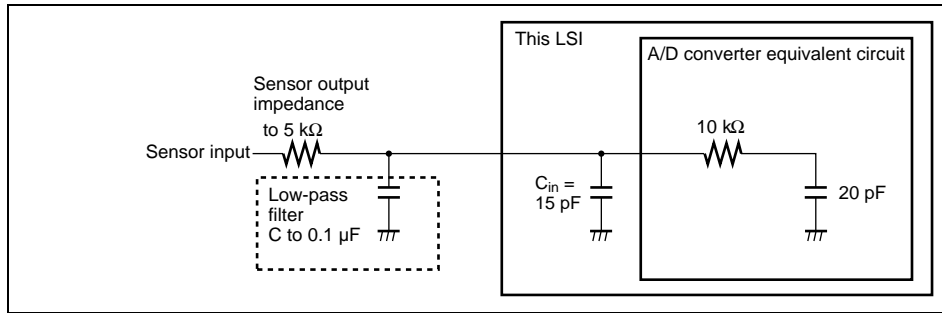


Figure 21.7 Example of Analog Input Circuit

The voltage applied to digital input pin CINn should be in the range $AV_{SS} \leq CINn \leq V_{SS}$ and $V_{SS} \leq CINn \leq V_{CC}$ ($n = 0$ to 15).

- Relation between AV_{CC} , AV_{SS} and V_{CC} , V_{SS}

For the relationship between AV_{CC} , AV_{SS} and V_{CC} , V_{SS} , set $AV_{SS} = V_{SS}$. If the A/D converter is not used, the AV_{CC} and AV_{SS} pins must on no account be left open.

- AV_{ref} pin reference voltage specification range

The reference voltage of the AV_{ref} pin should be in the range $AV_{ref} \leq AV_{CC}$.

21.7.4 Notes on Board Design

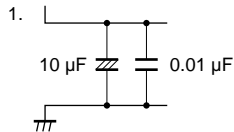
In board design, digital circuitry and analog circuitry should be as mutually isolated as possible and layout in which digital circuit signal lines and analog circuit signal lines cross or are in proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values. The digital circuitry must be isolated from the analog input signals ($AN0$ to $AN7$), analog reference voltage (AV_{ref}), and analog power supply (AV_{CC}) by the analog ground (AV_{SS}). Also, the analog ground (AV_{SS}) should be connected at one point to a stable digital ground (V_{SS}) on the board.

21.7.5 Notes on Noise Countermeasures

A protection circuit connected to prevent damage due to an abnormal voltage such as a surge at the analog input pins ($AN0$ to $AN7$) and analog reference voltage (AV_{ref}) should be connected between AV_{CC} and AV_{SS} as shown in figure 21.8. Also, the bypass capacitor should be connected to AV_{CC} and AV_{ref} , and the filter capacitor connected to $AN2$ to $AN7$, must be connected to AV_{SS} .

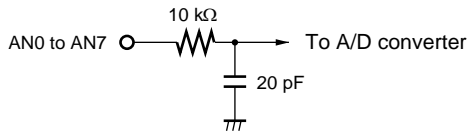
If a filter capacitor is connected, the input currents at the analog input pins ($AN0$ to $AN7$) are averaged, and so an error may arise. Also, when A/D conversion is performed frequently in scan mode, if the current charged and discharged by the capacitance of the sample-and-hold capacitor in the A/D converter exceeds the current input via the input impedance (R_{in}), an error will occur in the analog input pin voltage. Careful consideration is therefore required when deciding on constants.

Notes: Values are reference values.



2. R_{in} : Input impedance

Figure 21.8 Example of Analog Input Protection Circuit



Note: Values are reference values.

Figure 21.9 Equivalent Circuit of Analog Input Pin

21.7.6 Module Stop Mode Setting

A/D converter operation can be enabled or disabled using the module stop control register. The initial setting is for A/D converter operation to be halted. Register access is enabled by module stop mode. For details, refer to section 26, Power-Down Modes.

Product Classification		RAM Capacitance	RAM Address
Flash memory version	H8S/2161B	4 kbytes	H'E080–H'FFFF, H'FFF
	H8S/2160B	4 kbytes	H'E080–H'FFFF, H'FFF
	H8S/2141B	4 kbytes	H'E080–H'FFFF, H'FFF
	H8S/2140B	4 kbytes	H'E080–H'FFFF, H'FFF
	H8S/2145B	8 kbytes	H'D080–H'FFFF, H'FFF
	H8S/2148B	4 kbytes	H'E080–H'FFFF, H'FFF

- Size

Product Classification	RAM Capacitance	RAM Address
H8S/2161B	128 kbytes	H'000000–H'01FFFF (mode 3) H'0000–H'DFFF (mode 3)
H8S/2160B	64 kbytes	H'000000–H'00FFFF (mode 3) H'0000–H'DFFF (mode 3)
H8S/2141B	128 kbytes	H'000000–H'01FFFF (mode 3) H'0000–H'DFFF (mode 3)
H8S/2140B	64 kbytes	H'000000–H'00FFFF (mode 3) H'0000–H'DFFF (mode 3)
H8S/2145B	256 kbytes	H'000000–H'03FFFF (mode 3) H'0000–H'DFFF (mode 3)
H8S/2148B	128 kbytes	H'000000–H'01FFFF (mode 3) H'0000–H'DFFF (mode 3)

- Programming/erase methods

The flash memory is programmed 128 bytes at a time. Erase is performed in single

The flash memory is configured as follows:

- 64-kbyte version: 8 kbytes × 2 blocks, 16 kbytes × 1 block, 28 kbytes × 1 block, and 1 kbyte × 4 blocks
- 128-kbyte version: 32 kbytes × 2 blocks, 8 kbytes × 2 blocks, 16 kbytes × 1 block, and 1 kbyte × 4 blocks
- 256-kbyte version: 64 kbytes × 3 blocks, 32 kbytes × 1 block, and 4 kbytes × 8 blocks

To erase the entire flash memory, each block must be erased in turn.

- Programming/erase time

It takes 10 ms (typ.) to program the flash memory 128 bytes at a time; 80 μs (typ.)

Erasing one block takes 100 ms (typ.).

- Automatic bit rate adjustment
With data transfer in boot mode, this LSI's bit rate can be automatically adjusted to transfer bit rate of the host.
- Programming/erasing protection
Sets protection against flash memory programming/erasing via hardware, software, protection.
- Programmer mode
In addition to on-board programming mode, programmer mode is supported to program/erase the flash memory using a PROM programmer.

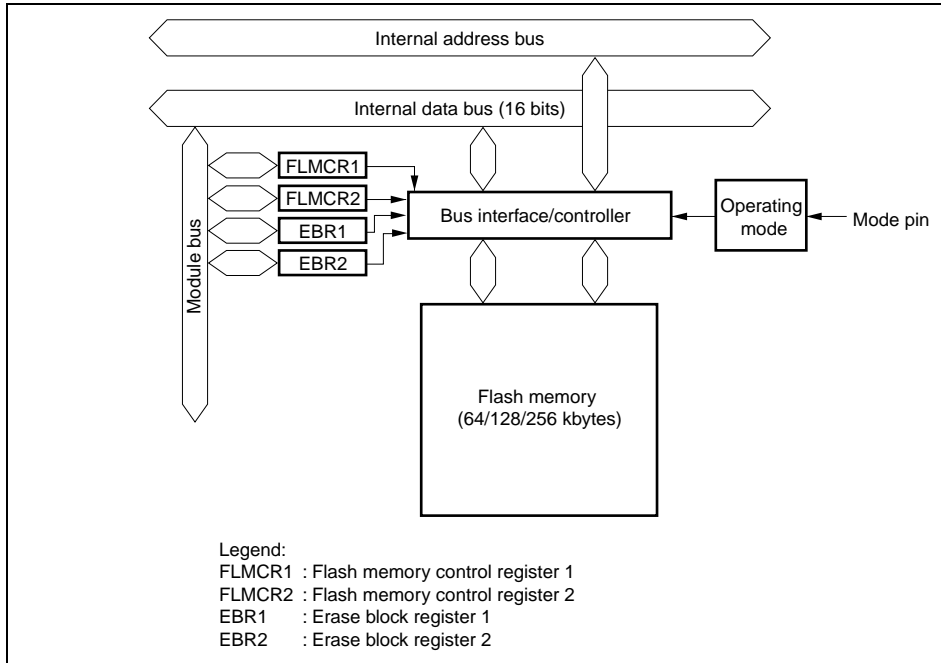


Figure 23.1 Block Diagram of Flash Memory

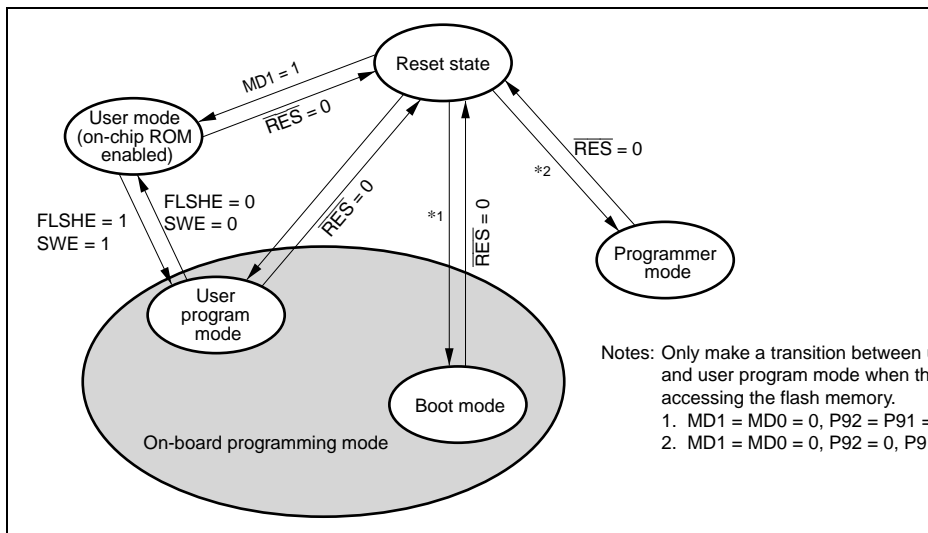
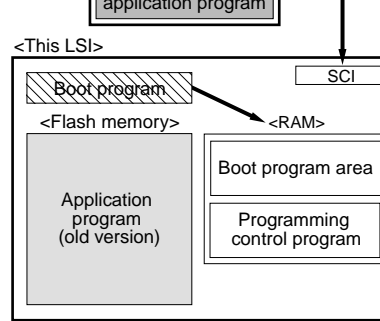
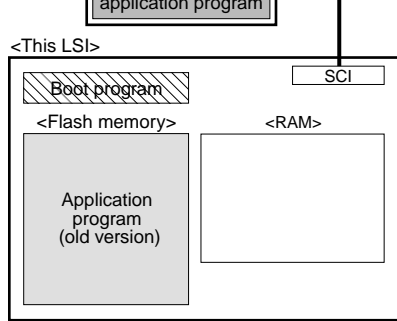


Figure 23.2 Flash Memory State Transitions

Table 23.1 Differences between Boot Mode and User Program Mode

	Boot Mode	User Program Mode
Total erase	Yes	Yes
Block erase	No	Yes
Programming control program*	Program/program-verify	Program/program-verify Erase/erase-verify

Note: * Should be provided by the user, in accordance with the recommended algorithm.

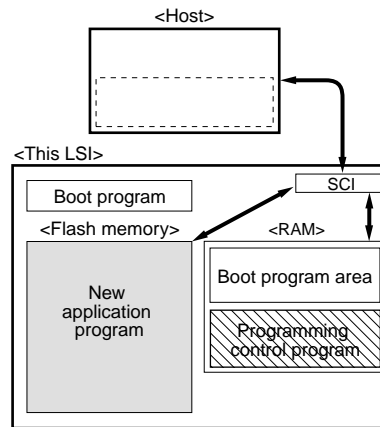
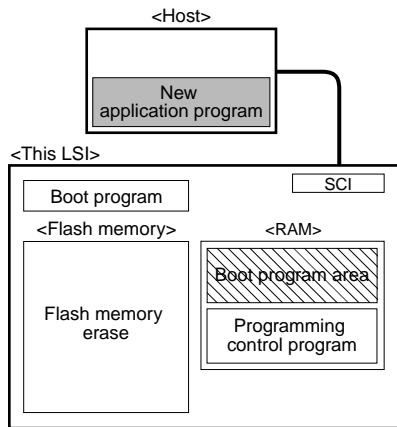


3. Flash memory initialization

The erase program in the boot program area (in RAM) is executed, and the flash memory is initialized (to H'FF). In boot mode, total flash memory erasure is performed, without regard to blocks.

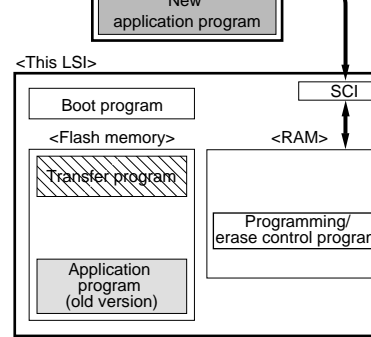
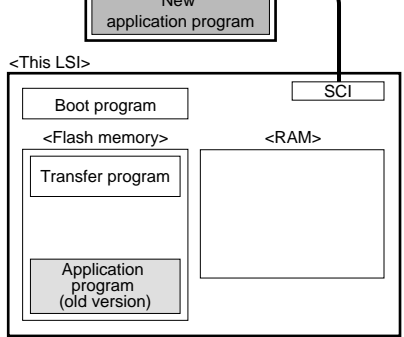
4. Writing new application program

The programming control program transfers the host to RAM via SCI communication is executed, and the new application program in the host is transferred into the flash memory.



Program execution

Figure 23.3 Boot Mode

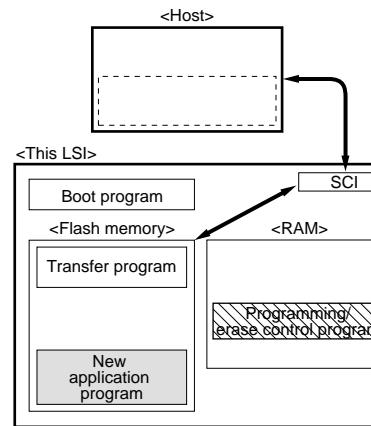
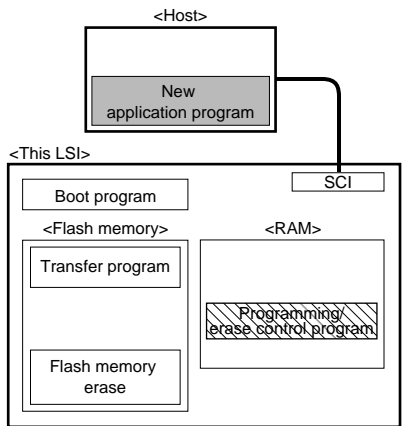


3. Flash memory initialization

The programming/erase program in RAM is executed, and the flash memory is initialized (to H'FF). Erasing can be performed in block units, but not in byte units.

4. Writing new application program

Next, the new application program in the host is written to the erased flash memory blocks. Do not write to unprogrammed blocks.




 Program execution

Figure 23.4 User Program Mode (Example)

units starting from an address whose lower bits are H'00 or H'80.

EB0 Erase unit: 1 kbyte	H'000000	H'000001	H'000002	← Programming unit: 128 bytes →
	H'000380	H'000381	H'000382	-----
EB1 Erase unit: 1 kbyte	H'000400	H'000401	H'000402	← Programming unit: 128 bytes →
	H'000780	H'000781	H'000782	-----
EB2 Erase unit: 1 kbyte	H'000800	H'000801	H'000802	← Programming unit: 128 bytes →
	H'000B80	H'000B81	H'000B82	-----
EB3 Erase unit: 1 kbyte	H'000C00	H'000C01	H'000C02	← Programming unit: 128 bytes →
	H'000F80	H'000F81	H'000F82	-----
EB4 Erase unit: 28 kbytes	H'001000	H'001001	H'001002	← Programming unit: 128 bytes →
	H'007F80	H'007F81	H'007F82	-----
EB5 Erase unit: 16 kbytes	H'008000	H'008001	H'008002	← Programming unit: 128 bytes →
	H'00BF80	H'00BF81	H'00BF82	-----
EB6 Erase unit: 8 kbytes	H'00C000	H'00C001	H'00C002	← Programming unit: 128 bytes →
	H'00DF80	H'00DF81	H'00DF82	-----
EB7 Erase unit: 8 kbytes	H'00E000	H'00E001	H'00E002	← Programming unit: 128 bytes →
	H'00FF80	H'00FF81	H'00FF82	-----

Figure 23.5 64-Kbyte Flash Memory Block Configuration

EB0 ↑ Erase unit: 1 kbyte ↓	H'000000	H'000001	H'000002	← Programming unit: 128 bytes →
	H'000380	H'000381	H'000382	-----
EB1 ↑ Erase unit: 1 kbyte ↓	H'000400	H'000401	H'000402	← Programming unit: 128 bytes →
	H'000780	H'000781	H'000782	-----
EB2 ↑ Erase unit: 1 kbyte ↓	H'000800	H'000801	H'000802	← Programming unit: 128 bytes →
	H'000B80	H'000B81	H'000B82	-----
EB3 ↑ Erase unit: 1 kbyte ↓	H'000C00	H'000C01	H'000C02	← Programming unit: 128 bytes →
	H'000F80	H'000F81	H'000F82	-----
EB4 ↑ Erase unit: 28 kbytes ↓	H'001000	H'001001	H'001002	← Programming unit: 128 bytes →
	H'007F80	H'007F81	H'007F82	-----
EB5 ↑ Erase unit: 16 kbytes ↓	H'008000	H'008001	H'008002	← Programming unit: 128 bytes →
	H'00BF80	H'00BF81	H'00BF82	-----
EB6 ↑ Erase unit: 8 kbytes ↓	H'00C000	H'00C001	H'00C002	← Programming unit: 128 bytes →
	H'00DF80	H'00DF81	H'00DF82	-----
EB7 ↑ Erase unit: 8 kbytes ↓	H'00E000	H'00E001	H'00E002	← Programming unit: 128 bytes →
	H'00FF80	H'00FF81	H'00FF82	-----
EB8 ↑ Erase unit: 32 kbytes ↓	H'010000	H'010001	H'010002	← Programming unit: 128 bytes →
	H'017F80	H'017F81	H'017F82	-----
EB9 ↑ Erase unit: 32 kbytes ↓	H'018000	H'018001	H'018002	← Programming unit: 128 bytes →
	H'01FF80	H'01FF81	H'01FF82	-----

Figure 23.6 128-Kbyte Flash Memory Block Configuration

EB0 Erase unit: 4 kbytes	H'000000	H'000001	H'000002	← Programming unit: 128 bytes →	H'000003
	H'000F80	H'000F81	H'000F82	-----	H'000F83
EB1 Erase unit: 4 kbytes	H'001000	H'001001	H'001002	← Programming unit: 128 bytes →	H'001003
	H'001F80	H'001F81	H'001F82	-----	H'001F83
EB2 Erase unit: 4 kbytes	H'002000	H'002001	H'002002	← Programming unit: 128 bytes →	H'002003
	H'002F80	H'002F81	H'002F82	-----	H'002F83
EB3 Erase unit: 4 kbytes	H'003000	H'003001	H'003002	← Programming unit: 128 bytes →	H'003003
	H'003F80	H'003F81	H'003F82	-----	H'003F83
EB4 Erase unit: 32 kbytes	H'004000	H'004001	H'004002	← Programming unit: 128 bytes →	H'004003
	H'00BF80	H'00BF81	H'00BF82	-----	H'00BF83
EB5 Erase unit: 4 kbytes	H'00C000	H'00C001	H'00C002	← Programming unit: 128 bytes →	H'00C003
	H'00CF80	H'00CF81	H'00CF82	-----	H'00CF83
EB6 Erase unit: 4 kbytes	H'00D000	H'00D001	H'00D002	← Programming unit: 128 bytes →	H'00D003
	H'00DF80	H'00DF81	H'00DF82	-----	H'00DF83
EB7 Erase unit: 4 kbytes	H'00E000	H'00E001	H'00E002	← Programming unit: 128 bytes →	H'00E003
	H'00EF80	H'00EF81	H'00EF82	-----	H'00EF83
EB8 Erase unit: 4 kbytes	H'00F000	H'00F001	H'00F002	← Programming unit: 128 bytes →	H'00F003
	H'00FF80	H'00FF81	H'00FF82	-----	H'00FF83
EB9 Erase unit: 64 kbytes	H'010000	H'010001	H'010002	← Programming unit: 128 bytes →	H'010003
	H'01FF80	H'01FF81	H'01FF82	-----	H'01FF83
EB10 Erase unit: 64 kbytes	H'020000	H'020001	H'020002	← Programming unit: 128 bytes →	H'020003
	H'02FF80	H'02FF81	H'02FF82	-----	H'02FF83
EB11 Erase unit: 64 kbytes	H'030000	H'030001	H'030002	← Programming unit: 128 bytes →	H'030003
	H'03FF80	H'03FF81	H'03FF82	-----	H'03FF83

Figure 23.7 256-Kbyte Flash Memory Block Configuration

MD0	Input	Sets this LSI's operating mode
P92	Input	Sets this LSI's operating mode
P91	Input	Sets this LSI's operating mode
P90	Input	Sets this LSI's operating mode
TxD1	Output	Serial transmit data output
RxD1	Input	Serial receive data input

23.5 Register Descriptions

The flash memory has the following registers. To access FLMCR1, FLMCR2, EBR1, the FLSHE bit in the serial/timer control register (STCR) should be set to 1. For details on the serial/timer control register, refer to section 3.2.3, Serial Timer Control Register (STCR).

- Flash memory control register 1 (FLMCR1)
- Flash memory control register 2 (FLMCR2)
- Erase block register 1 (EBR1)
- Erase block register 2 (EBR2)

23.5.1 Flash Memory Control Register 1 (FLMCR1)

FLMCR1, used together with FLMCR2, makes the flash memory transit to program mode, program-verify mode, erase mode, or erase-verify mode. For details on register settings, refer to section 23.8, Flash Memory Programming/Erasing.

FLMCR1 is initialized to H'80 by a reset, or in hardware standby mode, software standby mode, sub-active mode, sub-sleep mode, or watch mode.

cleared to 0, the EV, PV, E, and P bits in the
the ESU and PSU bits in FLMCR2, and all E
EBR2 bits cannot be set to 1. Do not clear th
and SWE to 0 simultaneously.

5, 4	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
3	EV	0	R/W	Erase-Verify When this bit is set to 1 while SWE = 1, the memory transits to erase-verify mode. When cleared to 0, erase-verify mode is cancelled.
2	PV	0	R/W	Program-Verify When this bit is set to 1 while SWE = 1, the memory transits to program-verify mode. When cleared to 0, program-verify mode is cancelled.
1	E	0	R/W	Erase When this bit is set to 1 while SWE = 1 and the flash memory transits to erase mode. When cleared to 0, erase mode is cancelled.
0	P	0	R/W	Program When this bit is set to 1 while SWE = 1 and the flash memory transits to program mode. When cleared to 0, program mode is cancelled.

7	FLER	0	R	Flash memory error Indicates that an error has occurred during flash memory programming/erasing. When the bit is set to 1, flash memory goes to the error-programming state. For details, see section 23.9.3, Error Programming.
6 to 2	—	All 0	R/(W)	Reserved The initial values should not be modified.
1	ESU	0	R/W	Erase Setup When this bit is set to 1 while SWE = 1, flash memory transits to the erase setup state. When the bit is cleared to 0, the erase setup state is cleared. Set this bit to 1 before setting the E bit in FL.
0	PSU	0	R/W	Program Setup When this bit is set to 1 while SWE = 1, flash memory transits to the program setup state. When the bit is cleared to 0, the program setup state is cleared. Set this bit to 1 before setting the P bit in FL to 1.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R/(W)	Reserved The initial values should not be modified.

- EBR2 (64-kbyte version)

Bit	Bit Name	Initial Value	R/W	Description
7	EB7	0	R/W*	When this bit is set to 1, 8 kbytes of EB7 (H'00E000 to H'00FFFF) are to be erased.
6	EB6	0	R/W	When this bit is set to 1, 8 kbytes of EB6 (H'00C000 to H'00DFFF) are to be erased.
5	EB5	0	R/W	When this bit is set to 1, 16 kbytes of EB5 (H'008000 to H'00BFFF) are to be erased.
4	EB4	0	R/W	When this bit is set to 1, 28 kbytes of EB4 (H'001000 to H'007FFF) are to be erased.
3	EB3	0	R/W	When this bit is set to 1, 1 kbyte of EB3 to H'000FFF) is to be erased.
2	EB2	0	R/W	When this bit is set to 1, 1 kbyte of EB2 to H'000BFF) is to be erased.
1	EB1	0	R/W	When this bit is set to 1, 1 kbyte of EB1 to H'0007FF) is to be erased.
0	EB0	0	R/W	When this bit is set to 1, 1 kbyte of EB0 to H'0003FF) is to be erased.

Note: * In normal mode, this bit is always read as 0 and cannot be modified.

- EBR2 (128-kbyte version)

Bit	Bit Name	Initial Value	R/W	Description
7	EB7	0	R/W*	When this bit is set to 1, 8 kbytes of EBR2 (H'00E000 to H'00FFFF) are to be erased.
6	EB6	0	R/W	When this bit is set to 1, 8 kbytes of EBR2 (H'00C000 to H'00DFFF) are to be erased.
5	EB5	0	R/W	When this bit is set to 1, 16 kbytes of EBR2 (H'008000 to H'00BFFF) are to be erased.
4	EB4	0	R/W	When this bit is set to 1, 28 kbytes of EBR2 (H'001000 to H'007FFF) are to be erased.
3	EB3	0	R/W	When this bit is set to 1, 1 kbyte of EBR2 (H'000000 to H'000FFF) is to be erased.
2	EB2	0	R/W	When this bit is set to 1, 1 kbyte of EBR2 (H'000000 to H'000BFF) is to be erased.
1	EB1	0	R/W	When this bit is set to 1, 1 kbyte of EBR2 (H'000000 to H'0007FF) is to be erased.
0	EB0	0	R/W	When this bit is set to 1, 1 kbyte of EBR2 (H'000000 to H'0003FF) is to be erased.

Note: * In normal mode, this bit is always read as 0 and cannot be modified.

1	EB9	0	R/W*	(H'020000 to H'02FFFF) are to be erased. When this bit is set to 1, 64 kbytes of EB9 (H'010000 to H'01FFFF) are to be erased.
0	EB8	0	R/W*	When this bit is set to 1, 4 kbytes of EB8 (H'000000 to H'00FFFF) are to be erased.

- EBR2 (256-kbyte version)

Bit	Bit Name	Initial Value	R/W	Description
7	EB7	0	R/W*	When this bit is set to 1, 4 kbytes of EB7 (H'00E000 to H'00EFFF) are to be erased.
6	EB6	0	R/W	When this bit is set to 1, 4 kbytes of EB6 (H'00D000 to H'00DFFF) are to be erased.
5	EB5	0	R/W	When this bit is set to 1, 4 kbytes of EB5 (H'00C000 to H'00CFFF) are to be erased.
4	EB4	0	R/W	When this bit is set to 1, 32 kbytes of EB4 (H'004000 to H'00BFFF) are to be erased.
3	EB3	0	R/W	When this bit is set to 1, 4 kbytes of EB3 (H'003000 to H'003FFF) are to be erased.
2	EB2	0	R/W	When this bit is set to 1, 4 kbytes of EB2 (H'002000 to H'002FFF) are to be erased.
1	EB1	0	R/W	When this bit is set to 1, 4 kbytes of EB1 (H'001000 to H'001FFF) are to be erased.
0	EB0	0	R/W	When this bit is set to 1, 4 kbytes of EB0 (H'000000 to H'000FFF) are to be erased.

Note: * In normal mode, this bit is always read as 0 and cannot be modified.

In normal mode (mode 3), up to 56 kbytes of ROM can be used.

Table 23.3 Operating Modes and ROM

MCU Operating Mode	Operating Modes		Mode Pins		MDCR
	CPU Operating Mode	Mode	MD1	MD0	EXPE
Mode 1	Normal	Expanded mode with on-chip ROM disabled	0	1	1
Mode 2	Advanced	Single-chip mode	1	0	0
	Advanced	Expanded mode with on-chip ROM enabled	1	0	1
Mode 3	Normal	Single-chip mode	1	1	0
	Normal	Expanded mode with on-chip ROM enabled	1	1	1

23.7 On-Board Programming Modes

An on-board programming mode is used to perform on-chip flash memory programming and verification. This LSI has two on-board programming modes: boot mode and user program mode. Table 23.4 shows pin settings for boot mode. In user program mode, operation is enabled by setting control bits. For details on flash memory mode transitions, see fi

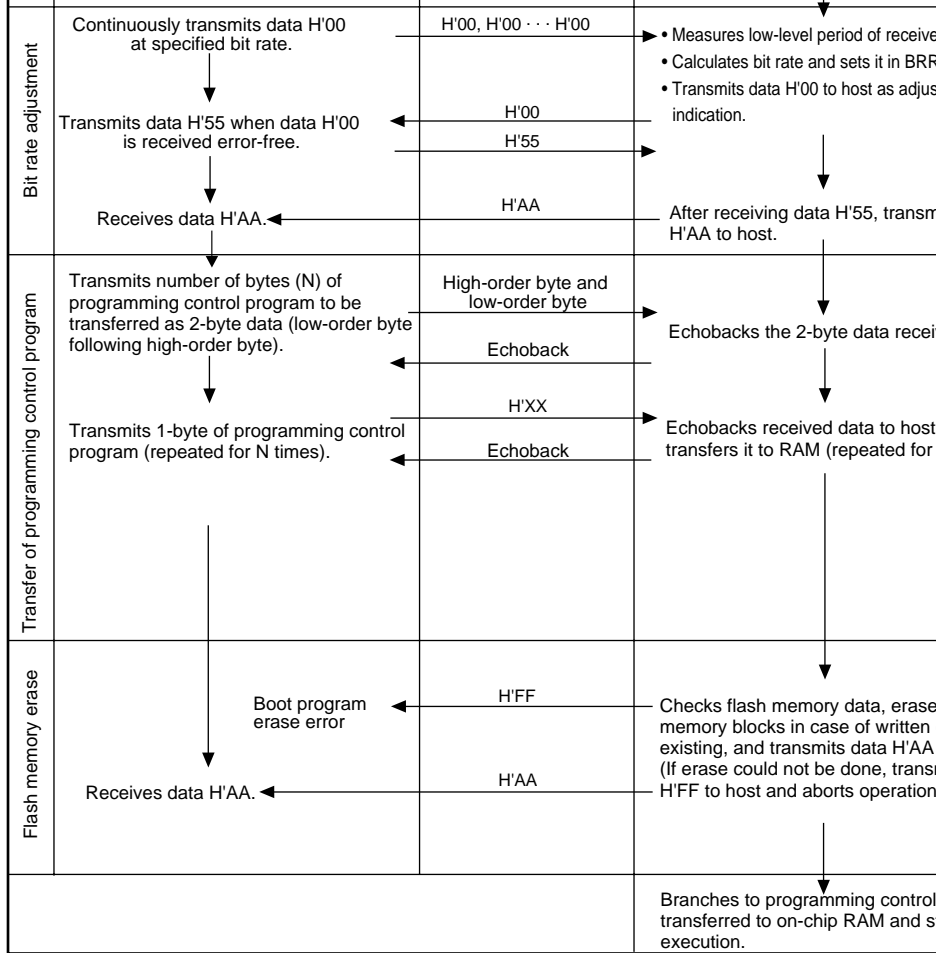
23.7.1 Boot Mode

Table 23.5 shows the boot mode operations between reset end and branching to the program control program.

1. When boot mode is used, the flash memory programming control program must be prepared by the host beforehand. Prepare a programming control program in accordance with the description in section 23.8, Flash Memory Programming/Erasing. In boot mode, if data exists in the flash memory (except in the case that all data are 1), all blocks in the flash memory are erased. Use boot mode at initial writing in the on-board state, or forced writing when user program mode cannot be executed because the program to be initiated in user program mode was mistakenly erased.
2. The SCI_1 should be set to asynchronous mode, and the transfer format as follows: 8 data bits, 1 stop bit, and no parity.
3. When the boot program is initiated, this LSI measures the low-level period of asynchronous SCI communication data (H'00) transmitted continuously from the host. This LSI then calculates the bit rate of transmission from the host, and adjusts the SCI_1 bit rate to match that of the host. The reset should end with the RxD1 pin high. The RxD1 and TxD1 pins should be pulled up on the board if necessary. After the reset ends, it takes approximately 100 μs before this LSI is ready to measure the low-level period.
4. After matching the bit rates, this LSI transmits one H'00 byte to the host to indicate the bit rate adjustment. The host should confirm that this adjustment end indication (H'00) has been received normally, and transmit one H'55 byte to this LSI. If reception could not be performed normally, initiate boot mode again by a reset. Depending on the host's transfer bit rate and system clock frequency of this LSI, there will be a discrepancy between the transfer bit rate of the host and this LSI. To operate the SCI properly, set the host's transfer bit rate to match the clock frequency of this LSI within the ranges listed in table 23.6.
5. In boot mode, a part of the on-chip RAM area is used by the boot program. Address H'FFE080 to H'FFE87F*¹ is the area to which the programming control program is transferred from the host. Note, however, that ID codes are assigned to addresses H'FFE080 to H'FFE087*². The boot program area cannot be used until the execution state in boot mode is reached.

- beginning of the programming control program, since the stack pointer (SP), in particular, is used implicitly in subroutine calls, etc.
7. Boot mode can be cleared by a reset. Cancel the reset*4 after driving the reset pin low for at least 20 states, and then setting the mode pins. Boot mode is also cleared when a program memory overflow occurs.
 8. Do not change the mode pin input levels in boot mode. If mode pin input levels are changed from low to high during reset, operating modes are switched and the state of ports used for address output and bus control output signals (\overline{AS} , \overline{RD} , and \overline{HWR}) are changed. Therefore, set these pins carefully not to be output signals during reset or not to coincide with LSI external signals.
 9. All interrupts are disabled during programming or erasing of the flash memory.

- Notes:
1. Address area for the H8S/2140B, H8S/2141B, H8S/2148B, H8S/2160B, and H8S/2161B. On the H8S/2145B, the address area is from H'FFD080 to H'FFD08F.
 2. Address area for the H8S/2140B, H8S/2141B, H8S/2148B, H8S/2160B, and H8S/2161B. On the H8S/2145B, the address area is from H'FFD080 to H'FFD08F.
 3. RAM address for the H8S/2140B, H8S/2141B, H8S/2148B, H8S/2160B, and H8S/2161B. On the H8S/2145B, the address is H'FFD088.
 4. After reset is cancelled, mode pin input settings must satisfy the mode pin setup time ($t_{MDS} = 4$ states).
 5. The ports that also have address output functions output low as address outputs when the mode pins are set to mode 1 during a reset. In modes other than mode 1, they enter the high impedance state. Bus control output signals output high when the mode pins are set to mode 1 during a reset. In modes other than mode 1, they enter the high impedance state.



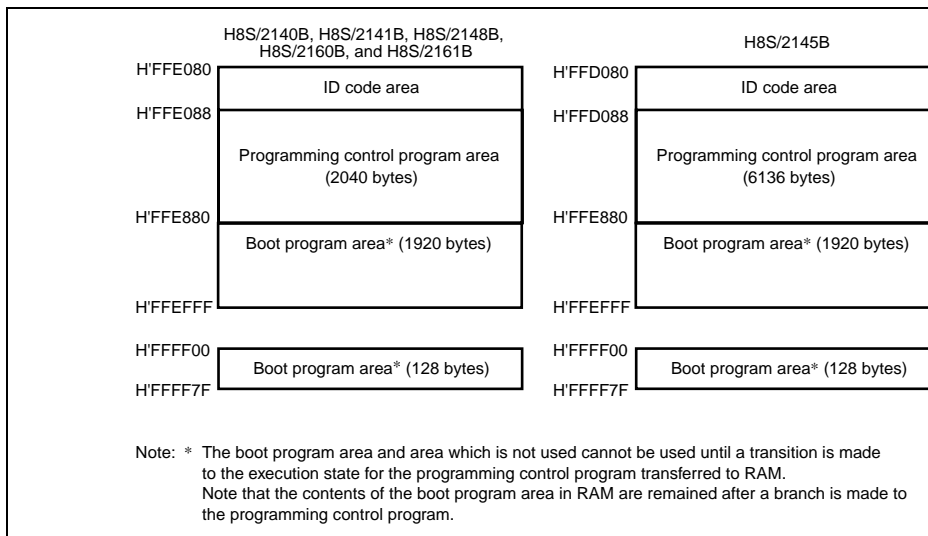


Figure 23.8 On-Chip RAM Area in Boot Mode

In boot mode, this LSI checks the contents of the 8-byte ID code area as shown below that the programming control program corresponds with this LSI. To originally write a programming control program to be used in boot mode, the above 8-byte ID code must be the beginning of the program.

Figure 23.9 ID Code Area

23.7.2 User Program Mode

On-board programming/erasing of an individual flash memory block can also be performed in user program mode by branching to a user program/erase control program. The user must set the necessary conditions and provide on-board means of supplying programming data. The flash memory blocks must contain the user program/erase control program or a program which provides the user program/erase control program from external memory. Because the flash memory itself cannot be read during programming/erasing, transfer the user program/erase control program to on-board RAM, as like in boot mode. Figure 23.10 shows a sample procedure for programming/erasing in user program mode. Prepare a user program/erase control program in accordance with the description in section 23.8, Flash Memory Programming/Erasing.

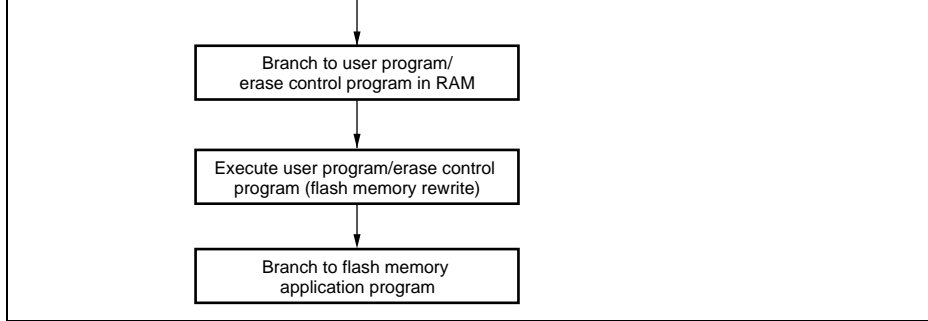
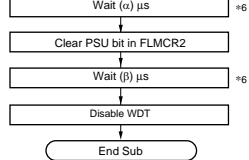


Figure 23.10 Programming/Erasing Flowchart Example in User Program

23.8 Flash Memory Programming/Erasing

A software method, using the CPU, is employed to program and erase flash memory in board programming modes. Depending on the FLMCR1 and FLMCR2 settings, the flash memory controller operates in one of the following four modes: program mode, program-verify mode, erase mode, and erase-verify mode. The programming control program in boot mode and the user program/erase control program in user program mode use these operating modes in combination to perform programming/erasing. Flash memory programming and erasing should be performed in accordance with the descriptions in section 23.8.1, Program/Program-Verify and section 23.8.2, Erase/Erase-Verify, respectively.

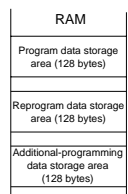
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer must be performed even if writing fewer than 128 bytes. In this case, H'FF data must be written to extra addresses.
3. Prepare the following data storage areas in RAM: a 128-byte programming data area, a 128-byte reprogramming data area, and a 128-byte additional-programming data area. Perform reprogramming data computation and additional programming data computation according to figure 23.11.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data area and additional-programming data area to the flash memory. The program address and 128 bytes of data are latched in the flash memory. The lower 8 bits of the start address in the flash memory destination area must be H'00 or H'80.
5. The time during which the P bit is set to 1 is the programming time. Figure 23.11 shows the allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program run-time error. The overflow cycle should be longer than $(y + z2 + \alpha + \beta)$ μ s.
7. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower 8 bits are B'00. Verify data can be read in words from the address to which a dummy write was performed.
8. The maximum number of repetitions of the program/program-verify sequence to the address is (N).



Note: 7. Write Pulse Width

Number of Writes n	Write Time (z) μs*6
1	z1
2	z1
3	z1
4	z1
5	z1
6	z1
7	z2
8	z2
9	z2
10	z2
11	z2
12	z2
13	z2
⋮	⋮
998	z2
999	z2
1000	z2

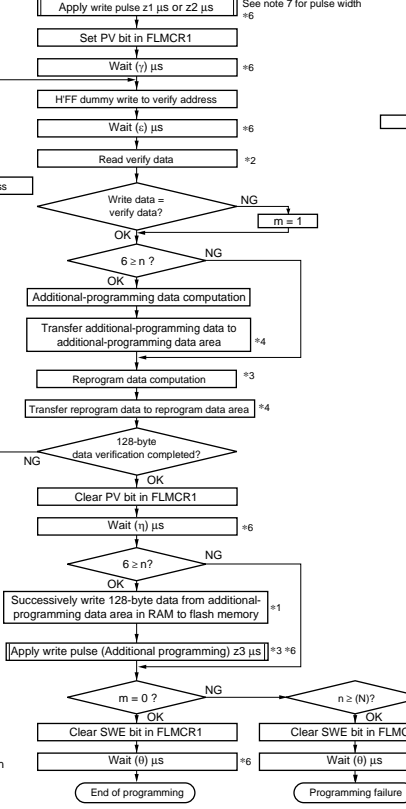
Note: Use a z3 μs write pulse for additional programming.



- Notes:
1. Data transfer is performed by byte transfer. The lower 8 bits of the first address written to must be H'00 or H'80. A 128-byte data transfer must be performed even if writing fewer than 128 bytes; in this case, HFF data must be written to the extra addresses.
 2. Verify data is read in 16-bit (word) units.
 3. Even bits for which programming has been completed will be subjected to programming once again if the result of the subsequent verify operation is NG.
 4. A 128-byte area for storing program data, a 128-byte area for storing reprogram data, and a 128-byte area for storing additional data must be provided in RAM. The contents of the reprogram data area and additional data area are modified as programming proceeds.
 5. A write pulse of z1 μs or z2 μs is applied according to the progress of the programming operation. See note 7 for details of the pulse widths. When writing of additional-programming data is executed, a z3 μs write pulse should be applied. Reprogram data X means reprogram data when the write pulse is applied.
 6. The values of x, y, z1, z2, z3, α, β, γ, α, η, θ, and N are shown in sections 28.1.6 and 28.2.6, Flash Memory Characteristics.

Reprogram Data Computation Table

Original Data (D)	Verify Data (V)	Reprogram Data (X)	Comments
0	0	1	Programming completed
0	1	0	Programming incomplete; reprogram
1	0	1	
1	1	1	Still in erased state; no action



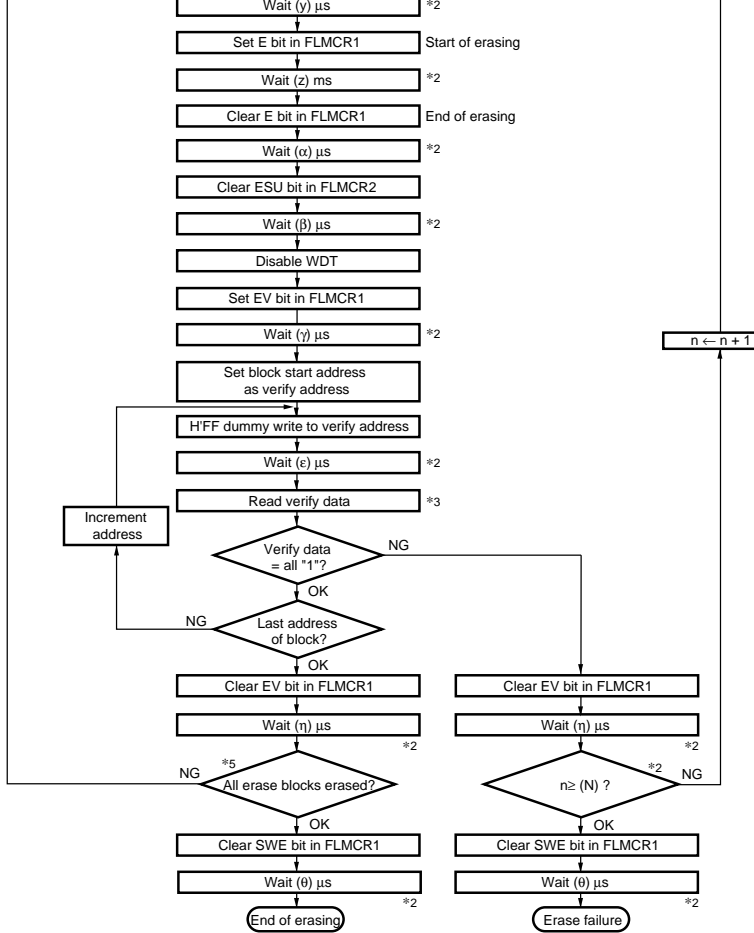
Additional-Programming Data Computation Table

Reprogram Data (X)	Verify Data (V)	Additional-Programming Data (Y)	Comments
0	0	0	
0	1	1	Additional programming to
1	0	1	Additional programming no
1	1	1	Additional programming no

Figure 23.11 Program/Program-Verify Flowchart



- turn.
3. The time during which the E bit is set to 1 is the flash memory erase time.
 4. The watchdog timer (WDT) is set to prevent overprogramming due to program runa. An overflow cycle of approximately $(y + z + \alpha + \beta)$ ms is allowed.
 5. For a dummy write to a verify address, write 1-byte data H'FF to an address whose bits are B'00. Verify data can be read in longwords from the address to which a dummy write was performed.
 6. If the read data is unerased, set erase mode again, and repeat the erase/erase-verify sequence before. The maximum number of repetitions of the erase/erase-verify sequence is N.



- Notes:
1. Prewriting (writing 0 to all data in erased block) is not necessary.
 2. The values of x, y, z, α, β, γ, ε, η, θ, and N are shown in sections 28.1.6 and 28.2.6, Flash Memory Characteristics.
 3. Verify data is read in 16-bit (word) units.
 4. Set only a single bit in EBR1 and EBR2. Do not set more than one bit.
 5. Erasing is performed in block units. To erase multiple blocks, each block must be erased in turn.

Figure 23.12 Erase/Erase-Verify Flowchart

or aborted by a reset (including WDT overflow reset), or a transition to hardware standby mode, software standby mode, sub-active mode, sub-sleep mode or watch mode. Flash memory registers 1 and 2 (FLMCR1 and FLMCR2) and erase block registers 1 and 2 (EBR1 and EBR2) are initialized. In a reset via the $\overline{\text{RES}}$ pin, the reset state is not entered unless the $\overline{\text{RES}}$ pin is held low until oscillation stabilizes after powering on. In the case of a reset during operation, the $\overline{\text{RES}}$ pin is held low for the $\overline{\text{RES}}$ pulse width specified in the AC Characteristics section.

23.9.2 Software Protection

Software protection can be implemented against programming/erasing of all flash memory by clearing the SWE bit in FLMCR1 to 0. When software protection is in effect, setting the SWE bit in FLMCR1 does not cause a transition to program mode or erase mode. By setting the EBR1 and EBR2 block registers 1 and 2 (EBR1 and EBR2), erase protection can be set for individual blocks. When EBR1 and EBR2 are set to H'00, erase protection is set for all blocks.

23.9.3 Error Protection

In error protection, an error is detected when the CPU's runaway occurs during flash memory programming/erasing, or operation is not performed in accordance with the program/erase algorithm, and the program/erase operation is aborted. Aborting the program/erase operation prevents damage to the flash memory due to overprogramming or overerasing.

When the following errors are detected during programming/erasing of flash memory, the ERR bit in FLMCR2 is set to 1, and the error protection state is entered.

- When the flash memory is read during programming/erasing (including vector reset instruction fetch)
- Immediately after exception handling (excluding a reset) during programming/erasing
- When a SLEEP instruction is executed (transits to software standby mode, sleep mode, sub-active mode, sub-sleep mode, or watch mode) during programming/erasing
- When the bus ownership is released during programming/erasing

In order to give the highest priority to programming/erasing operations, disable all interrupts including NMI input during flash memory programming/erasing (the P or E bit in FLMCR1 to 1) or boot program execution*¹.

1. If an interrupt is generated during programming/erasing, operation in accordance with the program/erase algorithm is not guaranteed.
2. CPU runaway may occur because normal vector reading cannot be performed in interrupt exception handling during programming/erasing*².
3. If an interrupt occurs during boot program execution, the normal boot mode sequence will not be executed.

Notes: 1. Interrupt requests must be disabled inside and outside the CPU until the program control program has completed programming.

2. The vector may not be read correctly for the following two reasons:

If flash memory is read while being programmed or erased (while the P or E bit in FLMCR1 is set to 1), correct read data will not be obtained (undefined data will be returned).

If the interrupt entry in the vector table has not been programmed yet, interrupt exception handling will not be executed correctly.

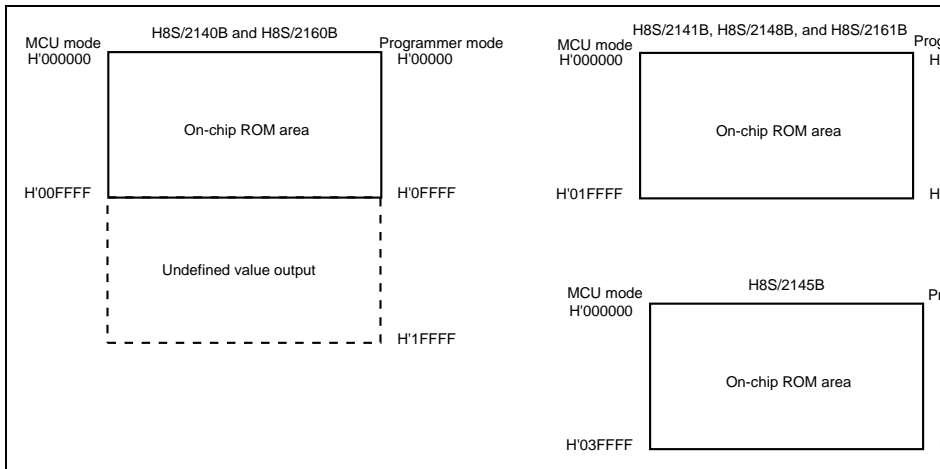


Figure 23.13 Memory Map in Programmer Mode

- Pin 261 (P1) of the programming voltage to 5.0 V.
2. Notes on power on/off
At powering on or off the Vcc power supply, fix the $\overline{\text{RES}}$ pin to low and set the flash memory to hardware protection state. This power on/off timing must also be satisfied at a power-on caused by a power failure and other factors.
 3. Perform flash memory programming/erasing in accordance with the recommended algorithm.
In the recommended algorithm, flash memory programming/erasing can be performed without subjecting this LSI to voltage stress or sacrificing program data reliability. When setting the SWE bit or E bit in FLMCR1 to 1, set the watchdog timer against program runaway.
 4. Do not set/clear the SWE bit during program execution in the flash memory.
Do not set/clear the SWE bit during program execution in the flash memory. An interval of at least 100 μs is necessary between program execution or data reading in flash memory and SWE bit clearing. When the SWE bit is set to 1, flash memory data can be modified. Flash memory data can be read only in program-verify or erase-verify mode. Do not read flash memory for a purpose other than verification during programming/erasing. Do not change the SWE bit during programming, erasing, or verifying.
 5. Do not use interrupts during flash memory programming/erasing
In order to give the highest priority to programming/erasing operation, disable all interrupts including NMI input when the flash memory is programmed or erased.
 6. Do not perform additional programming. Programming must be performed in the erase-verify mode.
Program the area with 128-byte programming-unit blocks in on-board programming mode only once. Perform programming in the state where the programming-unit block is fully erased.
 7. Ensure that the PROM programmer is correctly attached before programming.
If the socket, socket adapter, or product index does not match the specifications, to which current flows and the product may be damaged.
 8. Do not touch the socket adapter or LSI while programming.
Touching either of these can cause contact faults and write errors.

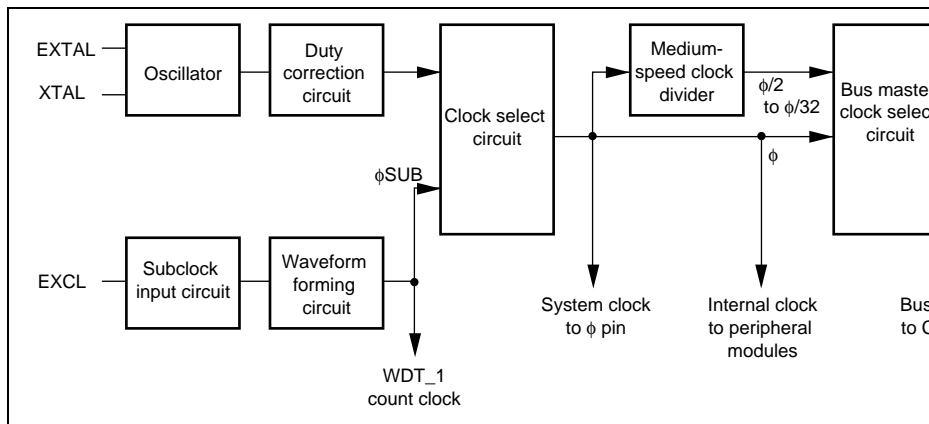


Figure 24.1 Block Diagram of Clock Pulse Generator

The bus master clock is selected as either high-speed mode or medium-speed mode by according to the settings of the SCK2 to SCK0 bits in the standby control register. For details on the standby control register, refer to section 26.1.1, Standby Control Register (SBYCR).

The subclock input is controlled by software according to the EXCLE bit setting in the low power control register. For details on the low power control register, refer to section 26.1.2, Low Power Control Register (LPWRCR).

resistance R_d , given in table 24.1, should be used. An A1-cut parallel-resonance crystal should be used.

Figure 24.3 shows the equivalent circuit of a crystal resonator. A resonator having the characteristics given in table 24.2 should be used.

A crystal resonator with frequency identical to that of the system clock (ϕ) should be used.

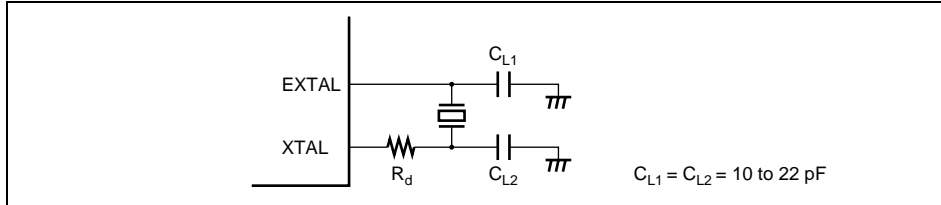


Figure 24.2 Typical Connection to Crystal Resonator

Table 24.1 Damping Resistance Values

Frequency (MHz)	2	4	8	10	12	16
R_d (Ω)	1 k	500	200	0	0	0

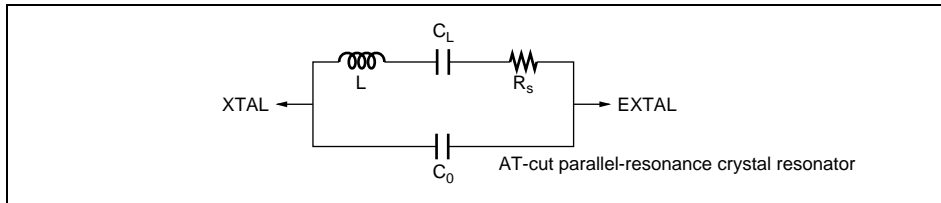


Figure 24.3 Equivalent Circuit of Crystal Resonator

Figure 24.4 shows a typical method of connecting an external clock signal. To leave the XTAL pin open, incidental capacitance should be 10 pF or less.

To input an inverted clock to the XTAL pin, the external clock should be set to high impedance mode, subactive mode, subsleep mode, and watch mode. External clock input conditions are shown in table 24.3. The frequency of the external clock should be the same as that of the internal clock (ϕ).

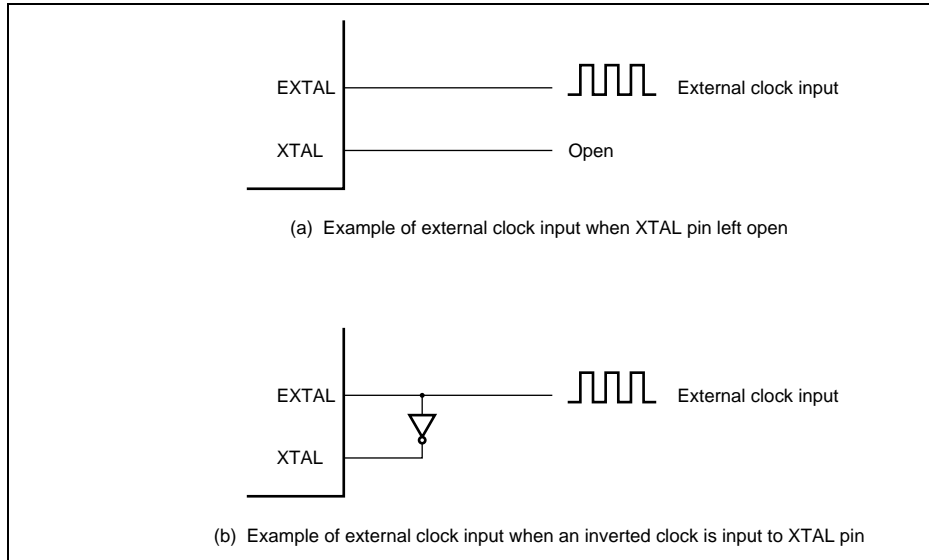


Figure 24.4 Example of External Clock Input

high level							
External clock rising time	t_{EXr}	—	10	—	5	ns	
External clock falling time	t_{EXf}	—	10	—	5	ns	
Clock pulse width low level	t_{CL}	0.4	0.6	0.4	0.6	t_{cyc}	$\phi \geq 5$ MHz
		80	—	80	—	ns	$\phi < 5$ MHz
Clock pulse width high level	t_{CH}	0.4	0.6	0.4	0.6	t_{cyc}	$\phi \geq 5$ MHz
		80	—	80	—	ns	$\phi < 5$ MHz

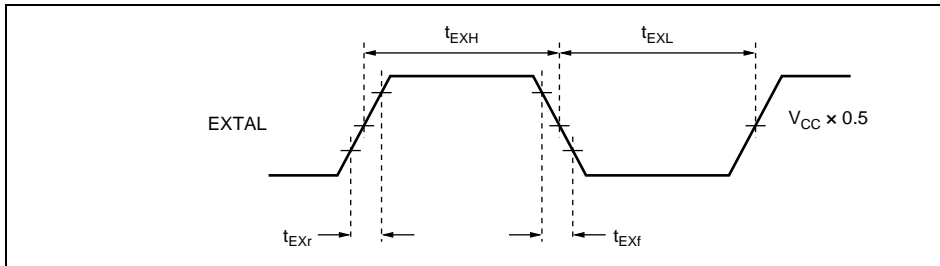


Figure 24.5 External Clock Input Timing

The oscillator and duty correction circuit have a function to adjust the waveform of the clock input that is input to the EXTAL pin. When a specified clock signal is input to the pin, internal clock signal output is determined after the external clock output stabilization time (t_{DEXT}) has passed. As the clock signal output is not determined during the t_{DEXT} cycle, signal should be set to low to hold it in reset state. Table 24.4 shows the external clock stabilization delay time. Figure 24.6 shows the timing of the external clock output stabilization delay time.

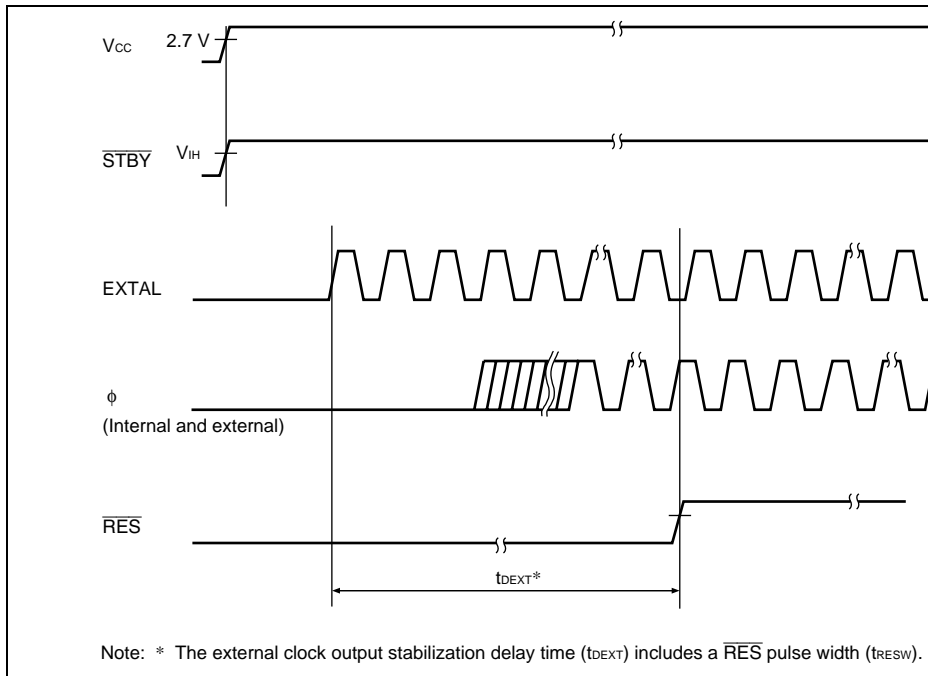


Figure 24.6 Timing of External Clock Output Stabilization Delay Time

24.2 Duty Correction Circuit

The duty correction circuit is valid when the oscillating frequency is 5 MHz or more. It corrects the duty of a clock that is output from the oscillator, and generates the system clock (φ).

24.3 Medium-Speed Clock Divider

The medium-speed clock divider divides the system clock (φ), and generates φ/2, φ/4, and φ/32 clocks.

The subclock input circuit controls subclock input from the EXCL pin.

Inputting the Subclock: To use the subclock, a 32.768-kHz external clock should be input to the EXCL pin. At this time, the P96DDR bit in P9DDR should be cleared to 0, and the subclock enable bit in LPWRCR should be set to 1.

Subclock input conditions are shown in table 24.5. When the subclock is not used, subclock input should not be enabled.

Table 24.5 Subclock Input Conditions

Item	Symbol	V _{CC} = 2.7 to 5.5 V			Unit	Meas Cond
		Min	Typ	Max		
Subclock input pulse width low level	t _{EXCLL}	—	15.26	—	μs	Figure 24.7
Subclock input pulse width high level	t _{EXCLH}	—	15.26	—	μs	
Subclock input rising time	t _{EXCLr}	—	—	10	ns	
Subclock input falling time	t _{EXCLf}	—	—	10	ns	

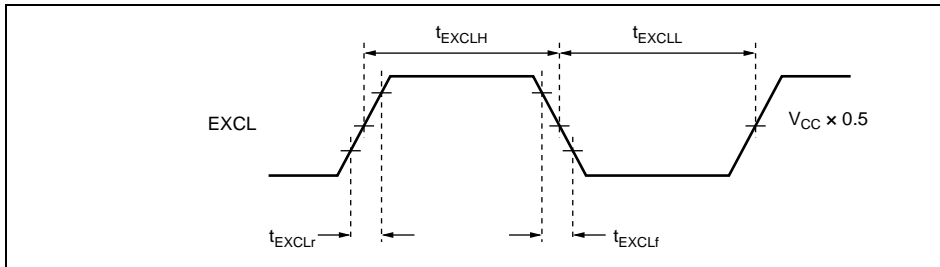


Figure 24.7 Subclock Input Timing

As described in the hardware manual (clock pulse generator/subclock input circuit), if the subclock is not used, the subclock input should not be enabled (EXCLE = 0).

24.6 Subclock Waveform Forming Circuit

To remove noise from the subclock input at the EXCL pin, the subclock is sampled by the system clock. The sampling frequency is set by the NESEL bit in LPWRCCR.

The subclock is not sampled in subactive mode, subsleep mode, or watch mode.

24.7 Clock Select Circuit

The clock select circuit selects the system clock that is used in this LSI.

A clock generated by an oscillator to which the EXTAL and XTAL pins are input is selected as the system clock when returning from high-speed mode, medium-speed mode, sleep mode, or standby mode.

A subclock input from the EXCL pin is selected as a system clock in subactive mode, subsleep mode, or watch mode. At this time, modules such as the CPU, TMR_0, TMR_1, WDT_0, WDT_1, ports, and interrupt controller and their functions operate depending on the system clock. The count clock and sampling clock for each timer are divided ϕ_{SUB} clocks.

Figure 24.8 Processing for X1 and X2 Pins

24.9 Usage Notes

24.9.1 Note on Resonator

Since all kinds of characteristics of the resonator are closely related to the board design, the user should use the example of resonator connection in this document for only reference; be sure to use an resonator that has been sufficiently evaluated by the user. Consult with the resonator manufacturer about the resonator circuit ratings which vary depending on the stray capacity of the resonator and installation circuit. Make sure the voltage applied to the oscillator pin does not exceed the maximum rating.

24.9.2 Notes on Board Design

When using a crystal resonator, the crystal resonator and its load capacitors should be placed as close as possible to the XTAL and EXTAL pins.

Other signal lines should be routed away from the oscillator circuit to prevent inductive interference with the correct oscillation as shown in figure 24.9.

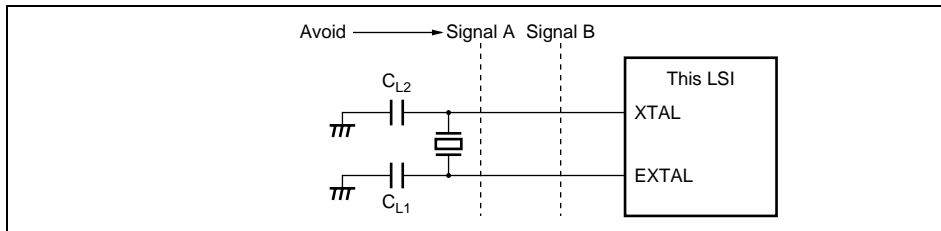


Figure 24.9 Note on Board Design of Oscillator Circuit Section

- System clock frequency for the CPU operation can be selected as $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$.
- Subactive mode
The CPU operates based on the subclock and on-chip peripheral modules other than TMR_0, TMR_1, WDT_0, and WDT_1 stop operating.
 - Sleep mode
The CPU stops but on-chip peripheral modules continue operating.
 - Subsleep mode
The CPU and on-chip peripheral modules other than TMR_0, TMR_1, WDT_0, and WDT_1 stop operating.
 - Watch mode
The CPU and on-chip peripheral modules other than WDT_1 stop operating.
 - Software standby mode
Clock oscillation stops, and the CPU and on-chip peripheral modules stop operating.
 - Hardware standby mode
Clock oscillation stops, and the CPU and on-chip peripheral modules enter reset state.
 - Module stop mode
Independently of above operating modes, on-chip peripheral modules that are not in reset state are stopped individually.

25.1 Register Descriptions

Power-down modes are controlled by the following registers. To access SBYCR, LPWRCR, MSTPCRH, and MSTPCRL, the FLSHE bit in the serial timer control register (STCR) must be cleared to 0. For details on STCR, see section 3.2.3, Serial Timer Control Register (STCR).

- Standby control register (SBYCR)
- Low power control register (LPWRCR)
- Module stop control register H (MSTPCRH)
- Module stop control register L (MSTPCRL)

when the SLEEP instruction is executed in high mode or medium-speed mode:

0: Shifts to sleep mode

1: Shifts to software standby mode, subactive mode or watch mode

When the SLEEP instruction is executed in subactive mode:

0: Shifts to subsleep mode

1: Shifts to watch mode or high-speed mode

Note that the SSBY bit is not changed even if a transition occurs by an interrupt.

6	STS2	0	R/W	Standby Timer Select 2 to 0
5	STS1	0	R/W	Selects the wait time for clock stabilization from oscillation start when canceling software standby mode, or subactive mode. Select a wait time (oscillation stabilization time) or more, depending on operating frequency. Table 25.1 shows the relationship between the STS2 to STS0 values and wait time. With an external clock, there are no specific wait time requirements. Normally the minimum value is recommended.
4	STS0	0	R/W	
3	—	0	R	Reserved This bit is always read as 0, and cannot be modified.

010: Medium-speed clock: $\phi/4$
 011: Medium-speed clock: $\phi/8$
 100: Medium-speed clock: $\phi/16$
 101: Medium-speed clock: $\phi/32$
 11X: —

Legend:

X: Don't care

Table 25.1 Operating Frequency and Wait Time

STS2	STS1	STS0	Wait Time	20 MHz	10 MHz	8 MHz	6 MHz	4 MHz	2 MHz
0	0	0	8192 states	0.4	0.8	1.0	1.3	2.0	4.1
0	0	1	16384 states	0.8	1.6	2.0	2.7	4.1	8.2
0	1	0	32768 states	2.0	3.3	4.1	5.5	8.2	16.4
0	1	1	65536 states	4.1	6.6	8.2	10.9	16.4	32.8
1	0	0	131072 states	8.2	13.1	16.4	21.8	32.8	65.5
1	0	1	262144 states	16.4	26.2	32.8	43.6	65.6	131.1
1	1	0	Reserved	—	—	—	—	—	—
1	1	1	16 states*	0.8	1.6	2.0	2.7	4.0	8.0

Shaded cells indicate the recommended specification.

Note: * This setting cannot be made in the flash-memory version of this LSI.

When the SLEEP instruction is executed in high-speed mode or medium-speed mode:

- 0: Shifts to sleep mode, software standby mode or software standby mode
- 1: Shifts directly to subactive mode, or shifts to software standby mode or software standby mode

When the SLEEP instruction is executed in subactive mode:

- 0: Shifts to subsleep mode or watch mode
- 1: Shifts directly to high-speed mode, or shifts to high-speed mode

6	LSON	0	R/W	<p>Low-Speed On Flag</p> <p>Specifies the operating mode to be entered after the SLEEP instruction. This bit also controls whether the processor shifts to high-speed mode or subactive mode when watch mode is cancelled.</p> <p>When the SLEEP instruction is executed in high-speed mode or medium-speed mode:</p> <ul style="list-style-type: none"> 0: Shifts to sleep mode, software standby mode or software standby mode 1: Shifts to watch mode or subactive mode <p>When the SLEEP instruction is executed in subactive mode:</p> <ul style="list-style-type: none"> 0: Shifts directly to watch mode or high-speed mode 1: Shifts to subsleep mode or watch mode <p>When watch mode is cancelled:</p> <ul style="list-style-type: none"> 0: Shifts to high-speed mode 1: Shifts to subactive mode
---	------	---	-----	--

4	EXCLE	0	R/W	Subclock Input Enable Enables/disables subclock input from the EXCL pin 0: Disables subclock input from the EXCL pin 1: Enables subclock input from the EXCL pin
3	—	0	R/W	Reserved An undefined value is read from this bit. This bit must be set to 1.
2 to 0	—	All 0	R	Reserved These bits are always read as 0 and cannot be set.

25.1.3 Module Stop Control Registers H and L (MSTPCRH, MSTPCRL)

MSTPCRH and MSTPCRL specify on-chip peripheral modules to shift to module stop mode. Each module can enter module stop mode by setting the corresponding bit.

- MSTPCRH

Bit	Bit Name	Initial Value	R/W	Corresponding Module
7	MSTP15	0*	R/W	
6	MSTP14	0	R/W	Data transfer controller (DTC)
5	MSTP13	1	R/W	16-bit free-running timer (FRT)
4	MSTP12	1	R/W	8-bit timers (TMR_0, TMR_1)
3	MSTP11	1	R/W	8-bit PWM timer (PWM), 14-bit PWM timer
2	MSTP10	1	R/W	D/A converter
1	MSTP9	1	R/W	A/D converter
0	MSTP8	1	R/W	8-bit timers (TMR_X, TMR_Y), timer counter

Note: * Do not set this bit to 1.

2	MSTP2	1	R/W	Host interface (XBS), keyboard buffer control register (KBCR), keyboard matrix interrupt mask register (KMIMR), keyboard matrix interrupt mask register A (KMIMRA), port 6 pull-up MOS control register (KMPCR)
1	MSTP1	1*	R/W	—
0	MSTP0	1	R/W	Host interface (LPC), wake-up event interrupt register B (WUEMRB)

Note: * This bit can be read from or written to, however, operation is not affected.

25.2 Mode Transitions and LSI States

Figure 25.1 shows the enabled mode transition diagram. The mode transition from program execution state to program halt state is performed by the SLEEP instruction. The mode transition from program halt state to program execution state is performed by an interrupt. The \overline{S} causes a mode transition from any state to hardware standby mode. The \overline{RES} input causes a transition from a state other than hardware standby mode to the reset state. Table 25.2 shows LSI internal states in each operating mode.

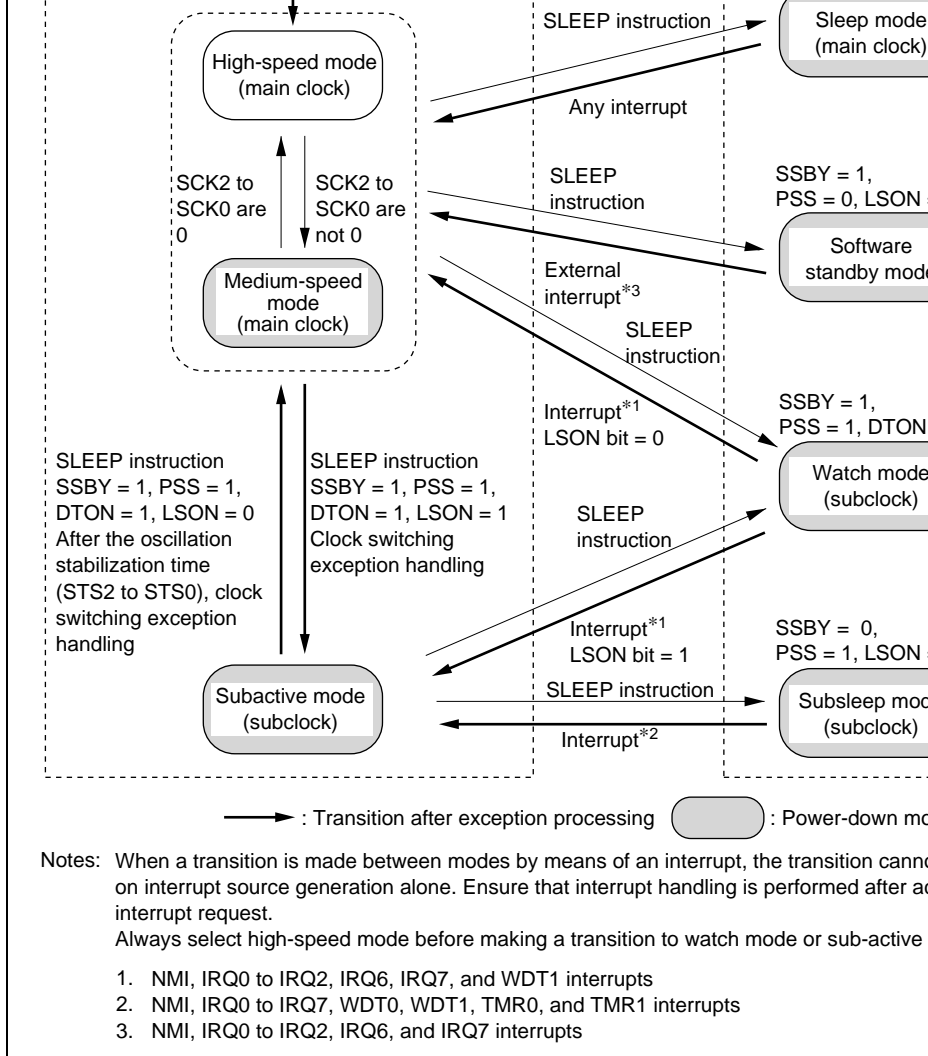


Figure 25.1 Mode Transition Diagram

External interrupts	NMI	Functioning	Functioning	Functioning	Functioning	Functioning	Functioning	Functioning	Functioning
	IRQ0 to IRQ7								
	KIN0 to KIN15								
	WUE0 to WUE7								
Peripheral modules	DTC	Functioning	Medium-speed operation	Functioning	Functioning/Halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)
	WDT_1	Functioning	Functioning	Functioning	Functioning	Subclock operation	Subclock operation	Subclock operation	Halted (retained)
	WDT_0	Functioning	Functioning	Functioning	Functioning	Halted (retained)	Subclock operation	Subclock operation	Halted (retained)
	TMR_0, TMR_1	Functioning	Functioning	Functioning	Functioning/Halted (retained)	Halted (retained)	Subclock operation	Subclock operation	Halted (retained)
	FRT	Functioning	Functioning	Functioning	Functioning/Halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)
	TMR_X, TMR_Y								
	Timer connection								
	IIC_0								
	IIC_1								
	LPC								
	SCI_0	Functioning	Functioning	Functioning	Functioning/Halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)
	SCI_1								
	SCI_2								
Peripheral modules	PWM	Functioning	Functioning	Functioning	Functioning/Halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)
	PWMX								
	XBS, Keyboard buffer controller								
	D/A								
	A/D								
	RAM	Functioning	Functioning	Functioning (DTC)	Functioning	Retained	Functioning	Retained	Retained
	I/O	Functioning	Functioning	Functioning	Functioning	Retained	Functioning	Functioning	Retained

Note: * "Halted (retained)" means that internal register values are retained. The internal state is "operation".
"Halted (reset)" means that internal register values and internal states are initialized.
In module stop mode, only modules for which a stop setting has been made are halted (reset or re

memory is accessed in 4 states, and internal I/O registers in 8 states.

By clearing all of bits SCK2 to SCK0 to 0, a transition is made to high-speed mode at the current bus cycle.

If a SLEEP instruction is executed when the SSBY bit in SBYCR is cleared to 0, and bit in LPWRCR is cleared to 0, a transition is made to sleep mode. When sleep mode is cancelled by an interrupt, medium-speed mode is restored. When the SLEEP instruction is executed with the SSBY bit set to 1, the LSON bit cleared to 0, and the PSS bit in TCSR (WDT_1) cleared to 0, operation shifts to software standby mode. When software standby mode is cleared by an interrupt, medium-speed mode is restored.

When the $\overline{\text{RES}}$ pin is set low and medium-speed mode is cancelled, operation shifts to hardware standby mode. The same applies in the case of a reset caused by overflow of the watchdog timer.

When the $\overline{\text{STBY}}$ pin is driven low, medium-speed mode is cancelled and a transition is made to hardware standby mode.

Figure 25.2 shows an example of medium-speed mode timing.

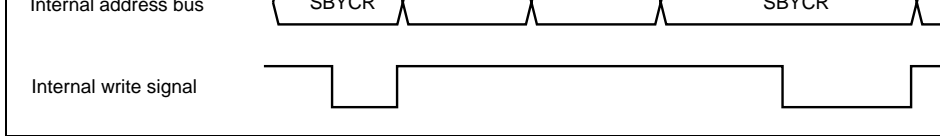


Figure 25.2 Medium-Speed Mode Timing

25.4 Sleep Mode

The CPU makes a transition to sleep mode if the SLEEP instruction is executed when the SLEEP bit in SBYCR is cleared to 0 and the LSON bit in LPWRCR is cleared to 0. In sleep mode, CPU operation stops but the peripheral modules do not stop. The contents of the CPU's internal registers are retained.

Sleep mode is exited by any interrupt, the $\overline{\text{RES}}$ pin, or the $\overline{\text{STBY}}$ pin.

When an interrupt occurs, sleep mode is exited and interrupt exception handling starts. Sleep mode is not exited if the interrupt is disabled, or interrupts other than NMI are masked by the interrupt mask register.

Setting the $\overline{\text{RES}}$ pin level low cancels sleep mode and selects the reset state. After the CPU stabilization time has passed, driving the $\overline{\text{RES}}$ pin high causes the CPU to start reset exception handling.

When the $\overline{\text{STBY}}$ pin level is driven low, sleep mode is cancelled and a transition is made to hardware standby mode.

states of on-chip peripheral modules other than the SCI, PWM, and PWMX, are retained as the prescribed voltage is supplied.

Software standby mode is cleared by an external interrupt (NMI, IRQ0 to IRQ2, IRQ6, or IRQ7), the $\overline{\text{RES}}$ pin input, or $\overline{\text{STBY}}$ pin input.

When an external interrupt request signal is input, system clock oscillation starts, and elapse of the time set in bits STS2 to STS0 in SBYCR, software standby mode is cleared and interrupt exception handling is started. When clearing software standby mode with an IRQ2, IRQ6, or IRQ7 interrupt, set the corresponding enable bit to 1 and ensure that no interrupt with a higher priority than interrupts IRQ0 to IRQ2, IRQ6, and IRQ7 is generated. Software standby mode cannot be cleared if an interrupt enable bit corresponding to an IRQ0 to IRQ2, IRQ6, or IRQ7 interrupt is cleared to 0 or if the interrupt has been masked on the CPU.

When the $\overline{\text{RES}}$ pin is driven low, system clock oscillation is started. At the same time as clock oscillation starts, the system clock is supplied to the entire LSI. Note that the $\overline{\text{RES}}$ pin be held low until clock oscillation stabilizes. When the $\overline{\text{RES}}$ pin goes high after clock oscillation stabilizes, the CPU begins reset exception handling.

When the $\overline{\text{STBY}}$ pin is driven low, software standby mode is cancelled and a transition to hardware standby mode.

Figure 25.3 shows an example in which a transition is made to software standby mode at the falling edge of the NMI pin, and software standby mode is cleared at the rising edge of the NMI pin.

In this example, an NMI interrupt is accepted with the NMIEG bit in SYSCR cleared (falling edge specification), then the NMIEG bit is set to 1 (rising edge specification), the SSB bit is set to 1, and a SLEEP instruction is executed, causing a transition to software standby mode.

Software standby mode is then cleared at the rising edge of the NMI pin.

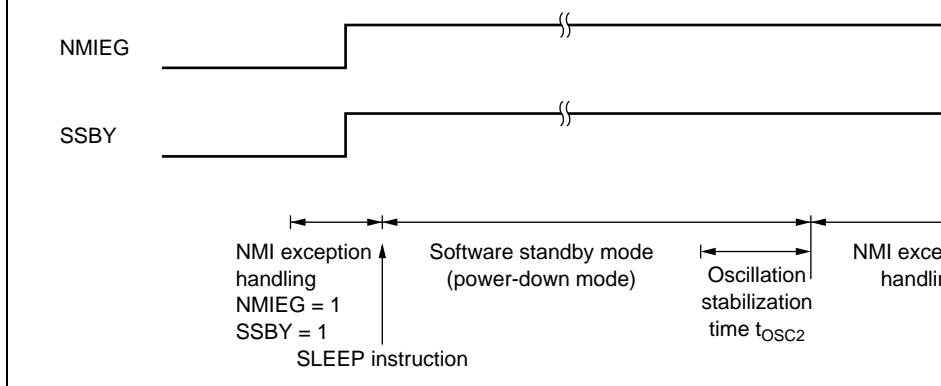


Figure 25.3 Application Example in Software Standby Mode

25.6 Hardware Standby Mode

The CPU makes a transition to hardware standby mode from any mode when the $\overline{\text{STBY}}$ pin is driven low.

In hardware standby mode, all functions enter the reset state. As long as the prescribed voltage is supplied, on-chip RAM data is retained. The I/O ports are set to the high-impedance state.

In order to retain on-chip RAM data, the RAME bit in SYSCR should be cleared to 0 before driving the $\overline{\text{STBY}}$ pin low. Do not change the state of the mode pins (MD1 and MD0) while the LSI is in hardware standby mode.

Hardware standby mode is cleared by the $\overline{\text{STBY}}$ pin input or the $\overline{\text{RES}}$ pin input.

When the $\overline{\text{STBY}}$ pin is driven high while the $\overline{\text{RES}}$ pin is low, clock oscillation is started. When the $\overline{\text{RES}}$ pin is held low until system clock oscillation stabilizes. When the $\overline{\text{RES}}$ pin is subsequently driven high after the clock oscillation stabilization time has passed, reset handling starts.

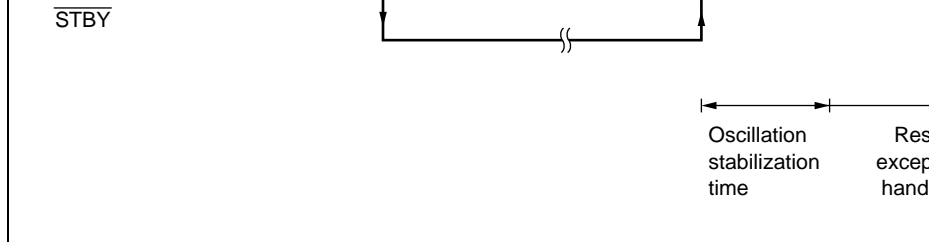


Figure 25.4 Hardware Standby Mode Timing

25.7 Watch Mode

The CPU makes a transition to watch mode when the SLEEP instruction is executed in high-speed mode or subactive mode with the SSBY bit in SBYCR set to 1, the DTON bit in LPWRCR cleared to 0, and the PSS bit in TCSR (WDT_1) set to 1.

In watch mode, the CPU is stopped and peripheral modules other than WDT_1 are also stopped. The contents of the CPU's internal registers, several on-chip peripheral module registers, and on-chip RAM data are retained and the I/O ports retain their values before transition as long as the prescribed voltage is supplied.

Watch mode is exited by an interrupt (WOVI1, NMI, IRQ0 to IRQ2, IRQ6, or IRQ7), a reset, the $\overline{\text{STBY}}$ pin input, or $\overline{\text{STBY}}$ pin input.

When an interrupt occurs, watch mode is exited and a transition is made to high-speed mode or medium-speed mode when the LSON bit in LPWRCR is cleared to 0 or to subactive mode when the LSON bit is set to 1. When a transition is made to high-speed mode, a stable clock is supplied to the entire LSI and interrupt exception handling starts after the time set in the STS2 to STS4 of SBYCR has elapsed. In the case of an IRQ0 to IRQ2, IRQ6, or IRQ7 interrupt, watch mode is exited if the corresponding enable bit has been cleared to 0. In the case of interrupts from other on-chip peripheral modules, watch mode is not exited if the interrupt enable register has been cleared to 0, disable the reception of that interrupt, or the interrupt is masked by the CPU.

The CPU makes a transition to subsleep mode when the SLEEP instruction is executed in subsleep mode with the SSBY bit in SBYCR cleared to 0, the LSON bit in LPWRCR cleared to 0, and the PSS bit in TCSR (WDT_1) set to 1.

In subsleep mode, the CPU is stopped. Peripheral modules other than TMR_0, TMR_1, and WDT_1 are also stopped. The contents of the CPU's internal registers, several on-chip peripheral module registers, and on-chip RAM data are retained and the I/O ports retain their values before transition as long as the prescribed voltage is supplied.

Subsleep mode is exited by an interrupt (interrupts by on-chip peripheral modules, NMI, IRQ7), the $\overline{\text{RES}}$ pin input, or the $\overline{\text{STBY}}$ pin input.

When an interrupt occurs, subsleep mode is exited and interrupt exception handling starts.

In the case of an IRQ0 to IRQ7 interrupt, subsleep mode is not exited if the corresponding bit has been cleared to 0. In the case of interrupts from the on-chip peripheral modules, subsleep mode is not exited if the interrupt enable register has been set to disable the reception of the interrupt, or the interrupt is masked by the CPU.

When the $\overline{\text{RES}}$ pin is driven low, system clock oscillation starts. Simultaneously with the start of system clock oscillation, the system clock is supplied to the entire LSI. Note that the $\overline{\text{RES}}$ pin must be held low until clock oscillation is stabilized. If the $\overline{\text{RES}}$ pin is driven high after the clock oscillation stabilization time has passed, the CPU begins reset exception handling.

If the $\overline{\text{STBY}}$ pin is driven low, the LSI enters hardware standby mode.

executes programs. Peripheral modules other than TMR_0, TMR_1, WDT_0, and WDT_1 are also stopped.

When operating the CPU in subactive mode, the SCK2 to SCK0 bits in SBYCR must be cleared to 0.

Subactive mode is exited by the SLEEP instruction, $\overline{\text{RES}}$ pin input, or $\overline{\text{STBY}}$ pin input.

When the SLEEP instruction is executed with the SSBY bit in SBYCR set to 1, the DTON bit in LPWRCR cleared to 0, and the PSS bit in TCSR (WDT_1) set to 1, the CPU exits subactive mode and a transition is made to watch mode. When the SLEEP instruction is executed with the SSBY bit in SBYCR cleared to 0, the LSON bit in LPWRCR set to 1, and the PSS bit in TCSR (WDT_1) set to 1, a transition is made to subsleep mode. When the SLEEP instruction is executed with the SSBY bit in SBYCR set to 1, the DTON bit and LSON bit in LPWRCR set to 10, and the PSS bit in TCSR (WDT_1) set to 1, a direct transition is made to high-speed mode.

For details of direct transitions, see section 25.11, Direct Transitions.

When the $\overline{\text{RES}}$ pin is driven low, system clock oscillation starts. Simultaneously with system clock oscillation, the system clock is supplied to the entire LSI. Note that the $\overline{\text{RES}}$ pin must be held low until the clock oscillation is stabilized. If the $\overline{\text{RES}}$ pin is driven high after system clock oscillation stabilization time has passed, the CPU begins reset exception handling.

If the $\overline{\text{STBY}}$ pin is driven low, the LSI enters hardware standby mode.

After the reset state is cancelled, all modules other than DTC are in module stop mode.

While an on-chip peripheral module is in module stop mode, read/write access to its registers is disabled.

25.11 Direct Transitions

The CPU executes programs in three modes: high-speed, medium-speed, and subactive mode. When a direct transition is made from high-speed mode to subactive mode, there is no interruption of program execution. A direct transition is enabled by setting the DTON bit in LPWRCR to 01, and then executing the SLEEP instruction. After a transition, direct transition exception handling starts.

The CPU makes a transition to subactive mode when the SLEEP instruction is executed in high-speed mode with the SSBY bit in SBYCR set to 1, the LSON bit and DTON bit in LPWRCR set to 01, and the PSS bit in TSCR (WDT_1) set to 1.

To make a direct transition to high-speed mode after the time set in the STS2 to STS0 bits in SBYCR has elapsed, execute the SLEEP instruction in subactive mode with the SSBY bit in SBYCR set to 1, the LSON bit and DTON bit in LPWRCR set to 01, and the PSS bit in TSCR (WDT_1) set to 1.

25.12.2 Current Consumption when Waiting for Oscillation Stabilization

The current consumption increases during oscillation stabilization.

25.12.3 DTC Module Stop Mode

If the DTC module stop mode specification and DTC bus request occur simultaneously, the DTC bus request is not released to the DTC and the MSTP bit cannot be set to 1. After completing the DTC bus request, the MSTP bit to 1 again.

- The MSB-side address is indicated for 16-bit addresses.
 - Registers are classified by functional modules.
 - The access size is indicated.
2. Register Bits
 - Bit configurations of the registers are described in the same order as the Register Addresses (address order) above.
 - Reserved bits are indicated by — in the bit name column.
 - The bit number in the bit-name column indicates that the whole register is allocated for counter or for holding data.
 - 16-bit registers are indicated from the bit on the MSB side.
 3. Register States in Each Operating Mode
 - Register states are described in the same order as the Register Addresses (address order) above.
 - The register states described here are for the basic operating modes. If there is a special mode for an on-chip peripheral module, refer to the section on that on-chip peripheral module.
 4. Register Select Conditions
 - Register states are described in the same order as the Register Addresses (address order) above.
 - For details on the register select conditions, refer to section 3.2.2, System Control Register (SYSCR), 3.2.3, Serial Timer Control Register (STCR), 26.1.3, Module Stop Control Register H and L (MSTPCRH, MSTPCRL), and the register descriptions for each module.

Register Name	Abbreviation	Of Bits	Address	Module	Width
Port G open drain control register	PGNOCR* ¹	8	H'FE16	PORT	8
Port E open drain control register	PENOCR* ¹	8	H'FE18	PORT	8
Port F open drain control register	PFNOCR* ¹	8	H'FE19	PORT	8
Port C open drain control register	PCNOCR* ¹	8	H'FE1C	PORT	8
Port D open drain control register	PDNOCR* ¹	8	H'FE1D	PORT	8
Bidirectional data register 0MW	TWR0MW	8	H'FE20	LPC	8
Bidirectional data register 0SW	TWR0SW	8	H'FE20	LPC	8
Bidirectional data register 1	TWR1	8	H'FE21	LPC	8
Bidirectional data register 2	TWR2	8	H'FE22	LPC	8
Bidirectional data register 3	TWR3	8	H'FE23	LPC	8
Bidirectional data register 4	TWR4	8	H'FE24	LPC	8
Bidirectional data register 5	TWR5	8	H'FE25	LPC	8
Bidirectional data register 6	TWR6	8	H'FE26	LPC	8
Bidirectional data register 7	TWR7	8	H'FE27	LPC	8
Bidirectional data register 8	TWR8	8	H'FE28	LPC	8
Bidirectional data register 9	TWR9	8	H'FE29	LPC	8
Bidirectional data register 10	TWR10	8	H'FE2A	LPC	8
Bidirectional data register 11	TWR11	8	H'FE2B	LPC	8
Bidirectional data register 12	TWR12	8	H'FE2C	LPC	8
Bidirectional data register 13	TWR13	8	H'FE2D	LPC	8
Bidirectional data register 14	TWR14	8	H'FE2E	LPC	8
Bidirectional data register 15	TWR15	8	H'FE2F	LPC	8
Input data register 3	IDR3	8	H'FE30	LPC	8
Output data register 3	ODR3	8	H'FE31	LPC	8
Status register 3	STR3	8	H'FE32	LPC	8

Input data register 1	IDR1	8	H'FE38	LPC	8
Output data register 1	ODR1	8	H'FE39	LPC	8
Status register 1	STR1	8	H'FE3A	LPC	8
Input data register 2	IDR2	8	H'FE3C	LPC	8
Output data register 2	ODR2	8	H'FE3D	LPC	8
Status register 2	STR2	8	H'FE3E	LPC	8
Host interface select register	HISEL	8	H'FE3F	LPC	8
Host interface control register 0	HICR0	8	H'FE40	LPC	8
Host interface control register 1	HICR1	8	H'FE41	LPC	8
Host interface control register 2	HICR2	8	H'FE42	LPC	8
Host interface control register 3	HICR3	8	H'FE43	LPC	8
Wakeup event interrupt mask register B	WUEMRB*2	8	H'FE44	INT	8
Port G output data register	PGODR*1	8	H'FE46	PORT	8
Port G input data register	PGPIN*1	8	H'FE47 (read)	PORT	8
Port G data direction register	PGDDR*1	8	H'FE47 (write)	PORT	8
Port E output data register	PEODR*1	8	H'FE48	PORT	8
Port F output data register	PFODR*1	8	H'FE49	PORT	8
Port E input data register	PEPIN*1	8	H'FE4A (read)	PORT	8
Port E data direction register	PEDDR*1	8	H'FE4A (write)	PORT	8
Port F input data register	PFPIN*1	8	H'FE4B (read)	PORT	8

Port C input data register	PCPIN ^{*1}	8	H'FE4E (read)	PORT	8
Port C data direction register	PCDDR ^{*1}	8	H'FE4E (write)	PORT	8
Port D input data register	PDPIN ^{*1}	8	H'FE4F (read)	PORT	8
Port D data direction register	PDDDR ^{*1}	8	H'FE4F (write)	PORT	8
Host interface control register 2	HICR2	8	H'FE80	XBS	8
Input data register_3	IDR_3	8	H'FE81	XBS	8
Output data register_3	ODR_3	8	H'FE82	XBS	8
Status register_3	STR_3	8	H'FE83	XBS	8
Input data register_4	IDR_4	8	H'FE84	XBS	8
Output data register_4	ODR_4	8	H'FE85	XBS	8
Status register_4	STR_4	8	H'FE86	XBS	8
I ² C bus extended control register_0	ICXR_0	8	H'FED4	IIC_0	8
I ² C bus extended control register_1	ICXR_1	8	H'FED5	IIC_1	8
Keyboard control register H_0	KBCRH_0	8	H'FED8	Keyboard buffer controller_0	8
Keyboard control register L_0	KBCRL_0	8	H'FED9	Keyboard buffer controller_0	8
Keyboard data buffer register_0	KBBR_0	8	H'FEDA	Keyboard buffer controller_0	8
Keyboard control register H_1	KBCRH_1	8	H'FEDC	Keyboard buffer controller_1	8

				controller_1	
Keyboard control register H_2	KBCRH_2	8	H'FEE0	Keyboard buffer controller_2	8
Keyboard control register L_2	KBCRL_2	8	H'FEE1	Keyboard buffer controller_2	8
Keyboard data buffer register_2	KBBR_2	8	H'FEE2	Keyboard buffer controller_2	8
Keyboard comparator control register	KBCOMP	8	H'FEE4	IrDA/Extended A/D	8
DDC switch register	DDCSWR	8	H'FEE6	IIC_0	8
Interrupt control register A	ICRA	8	H'FEE8	INT	8
Interrupt control register B	ICRB	8	H'FEE9	INT	8
Interrupt control register C	ICRC	8	H'FEEA	INT	8
IRQ status register	ISR	8	H'FEEB	INT	8
IRQ sense control register H	ISCRH	8	H'FEEC	INT	8
IRQ sense control register L	ISCR_L	8	H'FEED	INT	8
DTC enable register A	DTCERA	8	H'FEEE	DTC	8
DTC enable register B	DTCERB	8	H'FEEF	DTC	8
DTC enable register C	DTCERC	8	H'FEF0	DTC	8
DTC enable register D	DTCERD	8	H'FEF1	DTC	8
DTC enable register E	DTCERE	8	H'FEF2	DTC	8
DTC vector register	DTVECR	8	H'FEF3	DTC	8
Address break control register	ABRKCR	8	H'FEF4	INT	8

Flash memory control register 2	FLMCR2	8	H'FF81	FLASH	8
Peripheral clock select register	PCSR	8	H'FF82	PWM	8
Erase block register 1	EBR1	8	H'FF82	FLASH	8
System control register 2	SYSCR2	8	H'FF83	SYSTEM	8
Erase block register 2	EBR2	8	H'FF83	FLASH	8
Standby control register	SBYCR	8	H'FF84	SYSTEM	8
Low power control register	LPWRCR	8	H'FF85	SYSTEM	8
Module stop control register H	MSTPCRH	8	H'FF86	SYSTEM	8
Module stop control register L	MSTPCRL	8	H'FF87	SYSTEM	8
Serial mode register_1	SMR_1	8	H'FF88	SCI_1	8
I ² C bus control register_1	ICCR_1	8	H'FF88	IIC_1	8
Bit rate register_1	BRR_1	8	H'FF89	SCI_1	8
I ² C bus status register_1	ICSR_1	8	H'FF89	IIC_1	8
Serial control register_1	SCR_1	8	H'FF8A	SCI_1	8
Transmit data register_1	TDR_1	8	H'FF8B	SCI_1	8
Serial status register_1	SSR_1	8	H'FF8C	SCI_1	8
Receive data register_1	RDR_1	8	H'FF8D	SCI_1	8
Smart card mode register_1	SCMR_1	8	H'FF8E	SCI_1	8
I ² C bus data register_1	ICDR_1	8	H'FF8E	IIC_1	8
Second slave address register_1	SARX_1	8	H'FF8E	IIC_1	8
I ² C bus mode register_1	ICMR_1	8	H'FF8F	IIC_1	8
Slave address register_1	SAR_1	8	H'FF8F	IIC_1	8
Timer interrupt enable register	TIER	8	H'FF90	FRT	8
Timer control/status register	TCSR	8	H'FF91	FRT	8
Free running counter H	FRCH	8	H'FF92	FRT	8

Output control register BL	OCRBL	8	H'FF95	FRT	8
Timer control register	TCR	8	H'FF96	FRT	8
Timer output compare control register	TOCR	8	H'FF97	FRT	8
Input capture register AH	ICRAH	8	H'FF98	FRT	8
Output control register ARH	OCRARH	8	H'FF98	FRT	8
Input capture register AL	ICRAL	8	H'FF99	FRT	8
Output control register ARL	OCRARL	8	H'FF99	FRT	8
Input capture register BH	ICRBH	8	H'FF9A	FRT	8
Output control register AFH	OCRAFH	8	H'FF9A	FRT	8
Input capture register BL	ICRBL	8	H'FF9B	FRT	8
Output control register AFL	OCRAFL	8	H'FF9B	FRT	8
Input capture register CH	ICRCH	8	H'FF9C	FRT	8
Output compare register DMH	OCRDMH	8	H'FF9C	FRT	8
Input capture register CL	ICRCL	8	H'FF9D	FRT	8
Output compare register DML	OCRDML	8	H'FF9D	FRT	8
Input capture register DH	ICRDH	8	H'FF9E	FRT	8
Input capture register DL	ICRDL	8	H'FF9F	FRT	8
Serial mode register_2	SMR_2	8	H'FFA0	SCI_2	8
PWM (D/A) control register	DACR	8	H'FFA0	PWMX	8
PWM (D/A) data register AH	DADRAH	8	H'FFA0	PWMX	8
PWM (D/A) data register AL	DADRAL	8	H'FFA1	PWMX	8
Bit rate register_2	BRR_2	8	H'FFA1	SCI_2	8
Serial control register_2	SCR_2	8	H'FFA2	SCI_2	8
Transmit data register_2	TDR_2	8	H'FFA3	SCI_2	8

PWM (D/A) data register BH	DADRBH	8	H'FFA6	PWMX	8
PWM (D/A) counter L	DACNTL	8	H'FFA7	PWMX	8
PWM (D/A) data register BL	DADRBL	8	H'FFA7	PWMX	8
Timer control/status register_0	TCSR_0	8	H'FFA8	WDT	8
Timer counter_0	TCNT_0	8	H'FFA8 (write)	WDT_0	8
Timer counter_0	TCNT_0	8	H'FFA9 (read)	WDT_0	8
Port A output data register	PAODR	8	H'FFAA	PORT	8
Port A input data register	PAPIN	8	H'FFAB	PORT	8
Port A data direction register	PADDR	8	H'FFAB	PORT	8
Port 1 pull-up MOS control register	P1PCR	8	H'FFAC	PORT	8
Port 2 pull-up MOS control register	P2PCR	8	H'FFAD	PORT	8
Port 3 pull-up MOS control register	P3PCR	8	H'FFAE	PORT	8
Port 1 data direction register	P1DDR	8	H'FFB0	PORT	8
Port 2 data direction register	P2DDR	8	H'FFB1	PORT	8
Port 1 data register	P1DR	8	H'FFB2	PORT	8
Port 2 data register	P2DR	8	H'FFB3	PORT	8
Port 3 data direction register	P3DDR	8	H'FFB4	PORT	8
Port 4 data direction register	P4DDR	8	H'FFB5	PORT	8
Port 3 data register	P3DR	8	H'FFB6	PORT	8
Port 4 data register	P4DR	8	H'FFB7	PORT	8
Port 5 data direction register	P5DDR	8	H'FFB8	PORT	8
Port 6 data direction register	P6DDR	8	H'FFB9	PORT	8
Port 5 data register	P5DR	8	H'FFBA	PORT	8

Port 8 data direction register	P8DDR	8	H'FFBD (write)	PORT	8
Port 7 input data register	P7PIN	8	H'FFBE (read)	PORT	8
Port B data direction register	PBDDR	8	H'FFBE (write)	PORT	8
Port 8 data register	P8DR	8	H'FFBF	PORT	8
Port 9 data direction register	P9DDR	8	H'FFC0	PORT	8
Port 9 data register	P9DR	8	H'FFC1	PORT	8
Interrupt enable register	IER	8	H'FFC2	INT	8
Serial timer control register	STCR	8	H'FFC3	SYSTEM	8
System control register	SYSCR	8	H'FFC4	SYSTEM	8
Mode control register	MDCR	8	H'FFC5	SYSTEM	8
Bus control register	BCR	8	H'FFC6	BSC	8
Wait state control register	WSCR	8	H'FFC7	BSC	8
Timer control register_0	TCR_0	8	H'FFC8	TMR_0	8
Timer control register_1	TCR_1	8	H'FFC9	TMR_1	8
Timer control/status register_0	TCSR_0	8	H'FFCA	TMR_0	8
Timer control/status register_1	TCSR_1	8	H'FFCB	TMR_1	16
Time constant register A_0	TCORA_0	8	H'FFCC	TMR_0	16
Time constant register A_1	TCORA_1	8	H'FFCD	TMR_1	16
Time constant register B_0	TCORB_0	8	H'FFCE	TMR_0	16
Time constant register B_1	TCORB_1	8	H'FFCF	TMR_1	16
Timer counter_0	TCNT_0	8	H'FFD0	TMR_0	16
Timer counter_1	TCNT_1	8	H'FFD1	TMR_1	16
PWM output enable register B	PWOERB	8	H'FFD2	PWM	8

PWM data registers 0 to 15	PWDR0 to PWDR15	8	H'FFD7	PWM	8
Serial mode register_0	SMR_0	8	H'FFD8	SCI_0	8
I ² C bus control register_0	ICCR_0	8	H'FFD8	IIC_0	8
Bit rate register_0	BRR_0	8	H'FFD9	SCI_0	8
I ² C bus status register_0	ICSR_0	8	H'FFD9	IIC_0	8
Serial control register_0	SCR_0	8	H'FFDA	SCI_0	8
Transmit data register_0	TDR_0	8	H'FFDB	SCI_0	8
Serial status register_0	SSR_0	8	H'FFDC	SCI_0	8
Receive data register_0	RDR_0	8	H'FFDD	SCI_0	8
Smart card mode register_0	SCMR_0	8	H'FFDE	SCI_0	8
I ² C bus data register_0	ICDR_0	8	H'FFDE	IIC_0	8
Second slave address register_0	SARX_0	8	H'FFDE	IIC_0	8
I ² C bus mode register_0	ICMR_0	8	H'FFDF	IIC_0	8
Slave address register_0	SAR_0	8	H'FFDF	IIC_0	8
A/D data register AH	ADDRAH	8	H'FFE0	A/D converter	8
A/D data register AL	ADDRAL	8	H'FFE1	A/D converter	8
A/D data register BH	ADDRBH	8	H'FFE2	A/D converter	8
A/D data register BL	ADDRBL	8	H'FFE3	A/D converter	8
A/D data register CH	ADDRCH	8	H'FFE4	A/D converter	8
A/D data register CL	ADDRCL	8	H'FFE5	A/D converter	8
A/D data register DH	ADDRDH	8	H'FFE6	A/D converter	8
A/D data register DL	ADDRDL	8	H'FFE7	A/D converter	8
A/D control/status register	ADCSR	8	H'FFE8	A/D converter	8
A/D control register	ADCR	8	H'FFE9	A/D converter	8

Host interface control register	HICR	8	H'FFF0	XBS	8
Timer control register_X	TCR_X	8	H'FFF0	TMR_X	16
Timer control register_Y	TCR_Y	8	H'FFF0	TMR_Y	16
Keyboard matrix interrupt register 6	KMIMR	8	H'FFF1	INT	8
Timer control/status register_X	TCSR_X	8	H'FFF1	TMR_X	16
Timer control/status register_Y	TCSR_Y	8	H'FFF1	TMR_Y	16
Pull-up MOS control register	KMPCR	8	H'FFF2	PORT	8
Input capture register R	TICRR	8	H'FFF2	TMR_X	16
Time constant register A_Y	TCORA_Y	8	H'FFF2	TMR_Y	16
Keyboard matrix interrupt register A	KMIMRA	8	H'FFF3	INT	8
Input capture register F	TICRF	8	H'FFF3	TMR_X	16
Time constant register B_Y	TCORB_Y	8	H'FFF3	TMR_Y	16
Input data register_1	IDR_1	8	H'FFF4	XBS	8
Timer counter_X	TCNT_X	8	H'FFF4	TMR_X	16
Timer counter_Y	TCNT_Y	8	H'FFF4	TMR_Y	16
Output data register_1	ODR_1	8	H'FFF5	XBS	8
Timer constant register C	TCORC	8	H'FFF5	TMR_X	16
Timer input select register	TISR	8	H'FFF5	TMR_Y	16
Status register_1	STR_1	8	H'FFF6	XBS	8
Timer constant register A_X	TCORA_X	8	H'FFF6	TMR_X	16
Timer constant register B_X	TCORB_X	8	H'FFF7	TMR_X	16
D/A data register 0	DADR0	8	H'FFF8	D/A converter	8
D/A data register 1	DADR1	8	H'FFF9	D/A converter	8
D/A control register	DACR	8	H'FFFA	D/A converter	8
Input data register_2	IDR_2	8	H'FFFC	XBS	8

Status register_2	STR_2	8	H'FFFE	XBS	8
Timer connection register S	TCONRS	8	H'FFFE	Timer connection	8
Edge sense register	SEDGR	8	H'FFFF	Timer connection	8

- Notes: 1. Can be used on the H8S/2160B and H8S/2161B.
2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

PENOCR ^{*1}	PE7NOCR	PE6NOCR	PE5NOCR	PE4NOCR	PE3NOCR	PE2NOCR	PE1NOCR	PE0NOCR
PFNOCR ^{*1}	PF7NOCR	PF6NOCR	PF5NOCR	PF4NOCR	PF3NOCR	PF2NOCR	PF1NOCR	PF0NOCR
PCNOCR ^{*1}	PC7NOCR	PC6NOCR	PC5NOCR	PC4NOCR	PC3NOCR	PC2NOCR	PC1NOCR	PC0NOCR
PDNOCR ^{*1}	PD7NOCR	PD6NOCR	PD5NOCR	PD4NOCR	PD3NOCR	PD2NOCR	PD1NOCR	PD0NOCR
TWR0MW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR0SW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR3	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR4	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR5	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR6	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR7	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR9	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR10	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR11	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR12	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR13	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR14	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TWR15	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IDR3	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ODR3	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STR3 ^{*2}	IBF3B	OBF3B	MWMF	SWMF	C/D3	DBU32	IBF3A	OBF3A
STR3 ^{*3}	DBU37	DBU36	DBU35	DBU34	C/D3	DBU32	IBF3A	OBF3A
LADR3H	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
LADR3L	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	—	Bit 1	TWRE

ODR2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STR2	DBU27	DBU26	DBU25	DBU24	C/ \bar{D} 2	DBU22	IBF2	OBF2
HISEL	SELSTR3	SELIRQ11	SELIRQ10	SELIRQ9	SELIRQ6	SELSMI	SELIRQ12	SELIRQ
HICR0	LPC3E	LPC2E	LPC1E	FGA20E	SDWNE	PMEE	LSMIE	LSCIE
HICR1	LPCBSY	CLKREQ	IRQBSY	LRSTB	SDWNB	PMEB	LSMIB	LSCIB
HICR2	GA20	LRST	SDWN	ABRT	IBFIE3	IBFIE2	IBFIE1	ERRIE
HICR3	LFRAME	CLKRUN	SERIRQ	LRESET	LPCPD	PME	LSMI	LSCI
WUEMRB*5	WUEMR7	WUEMR6	WUEMR5	WUEMR4	WUEMR3	WUEMR2	WUEMR1	WUEMR0
PGODR*1	PG7ODR	PG6ODR	PG5ODR	PG4ODR	PG3ODR	PG2ODR	PG1ODR	PG0ODR
PGPIN*1	PG7PIN	PG6PIN	PG5PIN	PG4PIN	PG3PIN	PG2PIN	PG1PIN	PG0PIN
PGDDR*1	PG7DDR	PG6DDR	PG5DDR	PG4DDR	PG3DDR	PG2DDR	PG1DDR	PG0DDR
PEODR*1	PE7ODR	PE6ODR	PE5ODR	PE4ODR	PE3ODR	PE2ODR	PE1ODR	PE0ODR
PFODR*1	PF7ODR	PF6ODR	PF5ODR	PF4ODR	PF3ODR	PF2ODR	PF1ODR	PF0ODR
PEPIN*1	PE7PIN	PE6PIN	PE5PIN	PE4PIN	PE3PIN	PE2PIN	PE1PIN	PE0PIN
PEDDR*1	PE7DDR	PE6DDR	PE5DDR	PE4DDR	PE3DDR	PE2DDR	PE1DDR	PE0DDR
PFPIN*1	PF7PIN	PF6PIN	PF5PIN	PF4PIN	PF3PIN	PF2PIN	PF1PIN	PF0PIN
PFDDR*1	PF7DDR	PF6DDR	PF5DDR	PF4DDR	PF3DDR	PF2DDR	PF1DDR	PF0DDR
PCODR*1	PC7ODR	PC6ODR	PC5ODR	PC4ODR	PC3ODR	PC2ODR	PC1ODR	PC0ODR
PDODR*1	PD7ODR	PD6ODR	PD5ODR	PD4ODR	PD3ODR	PD2ODR	PD1ODR	PD0ODR
PCPIN*1	PC7PIN	PC6PIN	PC5PIN	PC4PIN	PC3PIN	PC2PIN	PC1PIN	PC0PIN
PCDDR*1	PC7DDR	PC6DDR	PC5DDR	PC4DDR	PC3DDR	PC2DDR	PC1DDR	PC0DDR
PDPIN*1	PD7PIN	PD6PIN	PD5PIN	PD4PIN	PD3PIN	PD2PIN	PD1PIN	PD0PIN
PDDDR*1	PD7DDR	PD6DDR	PD5DDR	PD4DDR	PD3DDR	PD2DDR	PD1DDR	PD0DDR
HICR2	—	—	—	—	—	IBFIE4	IBFIE3	—
IDR_3	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
ODR_3	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
STR_3	DBU	DBU	DBU	DBU	C/ \bar{D}	DBU	IBF	OBF

KBCRL_0	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0
KBBR_0	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0
KBCRH_1	KBIOE	KCLKI	KDI	KBFSEL	KBIE	KBF	PER	KBS
KBCRL_1	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0
KBBR_1	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0
KBCRH_2	KBIOE	KCLKI	KDI	KBFSEL	KBIE	KBF	PER	KBS
KBCRL_2	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0
KBBR_2	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0
KBCOMP	IrE	IrCKS2	IrCKS1	IrCKS0	KBADE	KBCH2	KBCH1	KBCH0

DDCSWR	SWE	SW	IE	IF	CLR3	CLR2	CLR1	CLR0
ICRA	ICRA7	ICRA6	ICRA5	ICRA4	ICRA3	ICRA2	ICRA1	ICRA0
ICRB	ICRB7	ICRB6	ICRB5	ICRB4	ICRB3	ICRB2	ICRB1	ICRB0
ICRC	ICRC7	ICRC6	ICRC5	ICRC4	ICRC3	ICRC2	ICRC1	ICRC0
ISR	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
ISCRH	IRQ7SCB	IRQ7SCA	IRQ6SCB	IRQ6SCA	IRQ5SCB	IRQ5SCA	IRQ4SCB	IRQ4SCA
ISCR_L	IRQ3SCB	IRQ3SCA	IRQ2SCB	IRQ2SCA	IRQ1SCB	IRQ1SCA	IRQ0SCB	IRQ0SCA
DTCEA	DTCEA7	DTCEA6	DTCEA5	DTCEA4	DTCEA3	DTCEA2	DTCEA1	DTCEA0
DTCEB	DTCEB7	DTCEB6	DTCEB5	DTCEB4	DTCEB3	DTCEB2	DTCEB1	DTCEB0
DTCEC	DTCEC7	DTCEC6	DTCEC5	DTCEC4	DTCEC3	DTCEC2	DTCEC1	DTCEC0
DTCED	DTCED7	DTCED6	DTCED5	DTCED4	DTCED3	DTCED2	DTCED1	DTCED0
DTCEE	DTCEE7	DTCEE6	DTCEE5	DTCEE4	DTCEE3	DTCEE2	DTCEE1	DTCEE0
DTVEC	DTVEC7	DTVEC6	DTVEC5	DTVEC4	DTVEC3	DTVEC2	DTVEC1	DTVEC0
ABRKCR	CMF	—	—	—	—	—	—	BIE
BARA	A23	A22	A21	A20	A19	A18	A17	A16
BARB	A15	A14	A13	A12	A11	A10	A9	A8

EBR2	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0
SBYCR	SSBY	STS2	STS1	STS0	—	SCK2	SCK1	SCK0
LPWRCR	DTON	LSON	NESEL	EXCLE	—	—	—	—
MSTPCRH	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	MSTP9	MSTP8
MSTPCRL	MSTP7	MSTP6	MSTP5	MSTP4	MSTP3	MSTP2	MSTP1	MSTP0
SMR_1	C \bar{A}	CHR	PE	O \bar{E}	STOP	MP	CKS1	CKS0
ICCR_1	ICE	IEIC	MST	TRS	ACKE	BBSY	IRIC	SCP
BRR_1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ICSR_1	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB
SCR_1	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
TDR_1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SSR_1	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
RDR_1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCMR_1	—	—	—	—	SDIR	SINV	—	SMIF
ICDR_1	ICDR7	ICDR6	ICDR5	ICDR4	ICDR3	ICDR2	ICDR1	ICDR0
SARX_1	SVAX6	SVAX5	SVAX4	SVAX3	SVAX2	SVAX1	SVAX0	FSX
ICMR_1	MLS	WAIT	CKS2	CKS1	CKS0	BC2	BC1	BC0
SAR_1	SVA6	SVA5	SVA4	SVA3	SVA2	SVA1	SVA0	FS
TIER	ICIAE	ICIBE	ICICE	ICIDE	OCIAE	OCIBE	OVIE	—
TCSR	ICFA	ICFB	ICFC	ICFD	OCFA	OCFB	OVF	CCLRA
FRCH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
FRCL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OCRAH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
OCRBH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
OCRAL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OCRBL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCR	IEDGA	IEDGB	IEDGC	IEDGD	BUFEA	BUFEB	CKS1	CKS0

OCRAFH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
ICRBL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OCRAFL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ICRCH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
OCRDMH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
ICRCL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OCRDML	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ICRDH	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
ICRDL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMR_2	C/ \bar{A}	CHR	PE	O/ \bar{E}	STOP	MP	CKS1	CKS0
DACR	TEST	PWME	—	—	OEB	OEA	OS	CKS
DADRAH	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6
DADRAL	DA5	DA4	DA3	DA2	DA1	DA0	CFS	—
BRR_2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCR_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
TDR_2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SSR_2	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
RDR_2	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCMR_2	—	—	—	—	SDIR	SINV	—	SMIF
DACNTH	UC7	UC6	UC5	UC4	UC3	UC2	UC1	UC0
DADRBH	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6
DACNTL	UC8	UC9	UC10	UC11	UC12	UC13	—	REGS
DADRBL	DA5	DA4	DA3	DA2	DA1	DA0	CFS	REGS
TCSR_0	OVF	WT/ \bar{I} T	TME	—	RST/ \bar{N} MI	CKS2	CKS1	CKS0
TCNT_0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PAODR	PA7ODR	PA6ODR	PA5ODR	PA4ODR	PA3ODR	PA2ODR	PA1ODR	PA0ODR
PAPIN	PA7PIN	PA6PIN	PA5PIN	PA4PIN	PA3PIN	PA2PIN	PA1PIN	PA0PIN

P1DR	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR	P10DR
P2DR	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR	P20DR
P3DDR	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR
P4DDR	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR
P3DR	P37DR	P36DR	P35DR	P34DR	P33DR	P32DR	P31DR	P30DR
P4DR	P47DR	P46DR	P45DR	P44DR	P43DR	P42DR	P41DR	P40DR
P5DDR	—	—	—	—	—	P52DDR	P51DDR	P50DDR
P6DDR	P67DDR	P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR
P5DR	—	—	—	—	—	P52DR	P51DR	P50DR
P6DR	P67DR	P66DR	P65DR	P64DR	P63DR	P62DR	P61DR	P60DR
PBODR	PB7ODR	PB6ODR	PB5ODR	PB4ODR	PB3ODR	PB2ODR	PB1ODR	PB0ODR
PBPIN	PB7PIN	PB6PIN	PB5PIN	PB4PIN	PB3PIN	PB2PIN	PB1PIN	PB0PIN
P8DDR	—	P86DDR	P85DDR	P84DDR	P83DDR	P82DDR	P81DDR	P80DDR
P7PIN	P77PIN	P76PIN	P75PIN	P74PIN	P73PIN	P72PIN	P71PIN	P70PIN
PBDDR	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR
P8DR	—	P86DR	P85DR	P84DR	P83DR	P82DR	P81DR	P80DR
P9DDR	P97DDR	P96DDR	P95DDR	P94DDR	P93DDR	P92DDR	P91DDR	P90DDR
P9DR	P97DR	P96DR	P95DR	P94DR	P93DR	P92DR	P91DR	P90DR
IER	IRQ7E	IRQ6E	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E
STCR	IICS	IICX1	IICX0	IICE	FLSHE	—	ICKS1	ICKS0
SYSCR	CS2E	IOSE	INTM1	INTM0	XRST	NMIEG	HIE	RAME
MDCR	EXPE	—	—	—	—	—	MDS1	MDS0
BCR	—	ICIS0	BRSTRM	BRSTS1	BRSTS0	—	IOS1	IOS0
WSCR	—	—	ABW	AST	WMS1	WMS0	WC1	WC0
TCR_0	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
TCR_1	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
TCSR_0	CMFB	CMFA	OVF	ADTE	OS3	OS2	OS1	OS0

TCNT_1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWOERB	OE15	OE14	OE13	OE12	OE11	OE10	OE9	OE8
PWOERA	OE7	OE6	OE5	OE4	OE3	OE2	OE1	OE0
PWDPRB	OS15	OS14	OS13	OS12	OS11	OS10	OS9	OS8
PWDpra	OS7	OS6	OS5	OS4	OS3	OS2	OS1	OS0
PWSL	PWCKE	PWCKS	—	—	RS3	RS2	RS1	RS0
PWDR0–15	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMR_0	C/ \bar{A}	CHR	PE	O/ \bar{E}	STOP	MP	CKS1	CKS0
ICCR_0	ICE	IEIC	MST	TRS	ACKE	BBSY	IRIC	SCP
BRR_0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ICSR_0	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB
SCR_0	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
TDR_0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SSR_0	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
RDR_0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCMR_0	—	—	—	—	SDIR	SINV	—	SMIF
ICDR_0	ICDR7	ICDR6	ICDR5	ICDR4	ICDR3	ICDR2	ICDR1	ICDR0
SARX_0	SVAX6	SVAX5	SVAX4	SVAX3	SVAX2	SVAX1	SVAX0	FSX
ICMR_0	MLS	WAIT	CKS2	CKS1	CKS0	BC2	BC1	BC0
SAR_0	SVA6	SVA5	SVA4	SVA3	SVA2	SVA1	SVA0	FS
ADDRAH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2
ADDRAL	AD1	AD0	—	—	—	—	—	—
ADDRBH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2
ADDRBL	AD1	AD0	—	—	—	—	—	—
ADDRCH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2
ADDRCL	AD1	AD0	—	—	—	—	—	—
ADDRDH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2

TCR_X	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
TCR_Y	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
KMIMR	KMIMR7	KMIMR6	KMIMR5	KMIMR4	KMIMR3	KMIMR2	KMIMR1	KMIMR0
TCSR_X	CMFB	CMFA	OVF	ICF	OS3	OS2	OS1	OS0
TCSR_Y	CMFB	CMFA	OVF	ICIE	OS3	OS2	OS1	OS0
KMPCR	KMIMR7	KMIMR6	KMIMR5	KMIMR4	KMIMR3	KMIMR2	KMIMR1	KMIMR0
TICRR	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCORA_Y	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
KMIMRA	KMIMR15	KMIMR14	KMIMR13	KMIMR12	KMIMR11	KMIMR10	KMIMR9	KMIMR8
TICRF	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCORB_Y	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IDR_1	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
TCNT_X	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCNT_Y	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ODR_1	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
TCORC	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TISR	—	—	—	—	—	—	—	IS
STR_1	DBU17	DBU16	DBU15	DBU14	C/D1	DBU12	IBF1	OBF1
TCORA_X	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCORB_X	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DADR_0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DADR_1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DACR	DAOE1	DAOE0	DAE	—	—	—	—	—
IDR_2	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
TCONRI	SIMOD1	SIMOD0	SCONE	ICST	HFINV	VFINV	HIINV	VIINV
ODR_2	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0

2. When TWRE = 1 or SELSTR3 = 0 in LADR3L
3. When TWRE = 0 and SELSTR3 = 1 in LADR3L
4. All bits are reserved in the 64-kbyte flash memory version.
The EB11 and EB10 bits are reserved in the 128-kbyte flash memory version.
5. Not supported by the H8S/2148B and H8S/2145B (5-V version).

PCNOCR ³¹	Initialized	—	—	—	—	—	—	—	—	Initialized
PDNOCR ³¹	Initialized	—	—	—	—	—	—	—	—	Initialized
TWR0MW	—	—	—	—	—	—	—	—	—	—
TWR0SW	—	—	—	—	—	—	—	—	—	—
TWR1	—	—	—	—	—	—	—	—	—	—
TWR2	—	—	—	—	—	—	—	—	—	—
TWR3	—	—	—	—	—	—	—	—	—	—
TWR4	—	—	—	—	—	—	—	—	—	—
TWR5	—	—	—	—	—	—	—	—	—	—
TWR6	—	—	—	—	—	—	—	—	—	—
TWR7	—	—	—	—	—	—	—	—	—	—
TWR8	—	—	—	—	—	—	—	—	—	—
TWR9	—	—	—	—	—	—	—	—	—	—
TWR10	—	—	—	—	—	—	—	—	—	—
TWR11	—	—	—	—	—	—	—	—	—	—
TWR12	—	—	—	—	—	—	—	—	—	—
TWR13	—	—	—	—	—	—	—	—	—	—
TWR14	—	—	—	—	—	—	—	—	—	—
TWR15	—	—	—	—	—	—	—	—	—	—
IDR3	—	—	—	—	—	—	—	—	—	—
ODR3	—	—	—	—	—	—	—	—	—	—
STR3	Initialized	—	—	—	—	—	—	—	—	Initialized
LADR3H	Initialized	—	—	—	—	—	—	—	—	Initialized
LADR3L	Initialized	—	—	—	—	—	—	—	—	Initialized
SIRQCR0	Initialized	—	—	—	—	—	—	—	—	Initialized
SIRQCR1	Initialized	—	—	—	—	—	—	—	—	Initialized
IDR1	—	—	—	—	—	—	—	—	—	—
ODR1	—	—	—	—	—	—	—	—	—	—



HICR0	Initialized	—	—	—	—	—	—	—	—	Initiali
HICR1	Initialized	—	—	—	—	—	—	—	—	Initiali
HICR2	Initialized	—	—	—	—	—	—	—	—	Initiali
HICR3	—	—	—	—	—	—	—	—	—	—
WUEMRB ^{*2}	Initialized	—	—	—	—	—	—	—	—	Initiali
PGODR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PGPIN ^{*1}	—	—	—	—	—	—	—	—	—	—
PGDDR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PEODR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PFODR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PEPIN ^{*1}	—	—	—	—	—	—	—	—	—	—
PEDDR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PFPIN ^{*1}	—	—	—	—	—	—	—	—	—	—
PFDDR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PCODR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PDODR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PCPIN ^{*1}	—	—	—	—	—	—	—	—	—	—
PCDDR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
PDPIN ^{*1}	—	—	—	—	—	—	—	—	—	—
PDDDR ^{*1}	Initialized	—	—	—	—	—	—	—	—	Initiali
HICR2	Initialized	—	—	—	—	—	—	—	—	Initiali
IDR_3	—	—	—	—	—	—	—	—	—	—
ODR_3	—	—	—	—	—	—	—	—	—	—
STR_3	Initialized	—	—	—	—	—	—	—	—	Initiali
IDR_4	—	—	—	—	—	—	—	—	—	Initiali
ODR_4	—	—	—	—	—	—	—	—	—	Initiali
STR_4	Initialized	—	—	—	—	—	—	—	—	Initiali

KBCRH_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBCRL_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBBR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBCRH_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBCRL_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBBR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
KBCOMP	Initialized	—	—	—	—	—	—	—	Initialized
<hr/>									
DDCSWR	Initialized	—	—	—	—	—	—	—	Initialized
ICRA	Initialized	—	—	—	—	—	—	—	Initialized
ICRB	Initialized	—	—	—	—	—	—	—	Initialized
ICRC	Initialized	—	—	—	—	—	—	—	Initialized
ISR	Initialized	—	—	—	—	—	—	—	Initialized
ISCRH	Initialized	—	—	—	—	—	—	—	Initialized
ISCR_L	Initialized	—	—	—	—	—	—	—	Initialized
DTCERA	Initialized	—	—	—	—	—	—	—	Initialized
DTCERB	Initialized	—	—	—	—	—	—	—	Initialized
DTCERC	Initialized	—	—	—	—	—	—	—	Initialized
DTCERD	Initialized	—	—	—	—	—	—	—	Initialized
DTCERE	Initialized	—	—	—	—	—	—	—	Initialized
DTVECR	Initialized	—	—	—	—	—	—	—	Initialized
ABRKCR	Initialized	—	—	—	—	—	—	—	Initialized
BARA	Initialized	—	—	—	—	—	—	—	Initialized
BARB	Initialized	—	—	—	—	—	—	—	Initialized
BARC	Initialized	—	—	—	—	—	—	—	Initialized
FLMCR1	Initialized	—	Initialized	—	Initialized	Initialized	—	Initialized	Initialized
FLMCR2	Initialized	—	Initialized	—	Initialized	Initialized	—	Initialized	Initialized



LPWRCR	Initialized	—	—	—	—	—	—	—	Initiali
MSTPCRH	Initialized	—	—	—	—	—	—	—	Initiali
MSTPCRL	Initialized	—	—	—	—	—	—	—	Initiali
SMR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
ICCR_1	Initialized	—	—	—	—	—	—	—	Initiali
BRR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
ICSR_1	Initialized	—	—	—	—	—	—	—	Initiali
SCR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
TDR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
SSR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
RDR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
SCMR_1	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initiali
ICDR_1	—	—	—	—	—	—	—	—	—
SARX_1	Initialized	—	—	—	—	—	—	—	Initiali
ICMR_1	Initialized	—	—	—	—	—	—	—	Initiali
SAR_1	Initialized	—	—	—	—	—	—	—	Initiali
TIER	Initialized	—	—	—	—	—	—	—	Initiali
TCSR	Initialized	—	—	—	—	—	—	—	Initiali
FRCH	Initialized	—	—	—	—	—	—	—	Initiali
FRCL	Initialized	—	—	—	—	—	—	—	Initiali
OCRAH	Initialized	—	—	—	—	—	—	—	Initiali
OCRBH	Initialized	—	—	—	—	—	—	—	Initiali
OCRAL	Initialized	—	—	—	—	—	—	—	Initiali
OCRBL	Initialized	—	—	—	—	—	—	—	Initiali
TCR	Initialized	—	—	—	—	—	—	—	Initiali
TOCR	Initialized	—	—	—	—	—	—	—	Initiali
ICRAH	Initialized	—	—	—	—	—	—	—	Initiali
OCRARH	Initialized	—	—	—	—	—	—	—	Initiali



OCRAFL	Initialized	—	—	—	—	—	—	—	Initialized
ICRCH	Initialized	—	—	—	—	—	—	—	Initialized
OCRDMH	Initialized	—	—	—	—	—	—	—	Initialized
ICRCL	Initialized	—	—	—	—	—	—	—	Initialized
OCRDM L	Initialized	—	—	—	—	—	—	—	Initialized
ICRDH	Initialized	—	—	—	—	—	—	—	Initialized
ICRDL	Initialized	—	—	—	—	—	—	—	Initialized
SMR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DACR	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DADRAH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DADRAL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
BRR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
SCR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
TDR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
SSR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
RDR_2	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
SCMR_2	Initialized	—	—	—	—	—	—	—	Initialized
DACNTH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DADRBH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DACNTL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
DADRBL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized
TCSR_0	Initialized	—	—	—	—	—	—	—	Initialized
TCNT_0	Initialized	—	—	—	—	—	—	—	Initialized
PAODR	Initialized	—	—	—	—	—	—	—	Initialized
PAPIN	—	—	—	—	—	—	—	—	—
PADDR	Initialized	—	—	—	—	—	—	—	Initialized
P1PCR	Initialized	—	—	—	—	—	—	—	Initialized
P2PCR	Initialized	—	—	—	—	—	—	—	Initialized



P3DDR	Initialized	—	—	—	—	—	—	—	Initiali
P4DDR	Initialized	—	—	—	—	—	—	—	Initiali
P3DR	Initialized	—	—	—	—	—	—	—	Initiali
P4DR	Initialized	—	—	—	—	—	—	—	Initiali
P5DDR	Initialized	—	—	—	—	—	—	—	Initiali
P6DDR	Initialized	—	—	—	—	—	—	—	Initiali
P5DR	Initialized	—	—	—	—	—	—	—	Initiali
P6DR	Initialized	—	—	—	—	—	—	—	Initiali
PBODR	Initialized	—	—	—	—	—	—	—	Initiali
PBPIN	—	—	—	—	—	—	—	—	—
P8DDR	Initialized	—	—	—	—	—	—	—	Initiali
P7PIN	—	—	—	—	—	—	—	—	—
PBDDR	Initialized	—	—	—	—	—	—	—	Initiali
P8DR	Initialized	—	—	—	—	—	—	—	Initiali
P9DDR	Initialized	—	—	—	—	—	—	—	Initiali
P9DR	Initialized	—	—	—	—	—	—	—	Initiali
IER	Initialized	—	—	—	—	—	—	—	Initiali
STCR	Initialized	—	—	—	—	—	—	—	Initiali
SYSCR	Initialized	—	—	—	—	—	—	—	Initiali
MDCR	Initialized	—	—	—	—	—	—	—	Initiali
BCR	Initialized	—	—	—	—	—	—	—	Initiali
WSCR	Initialized	—	—	—	—	—	—	—	Initiali
TCR_0	Initialized	—	—	—	—	—	—	—	Initiali
TCR_1	Initialized	—	—	—	—	—	—	—	Initiali
TCSR_0	Initialized	—	—	—	—	—	—	—	Initiali
TCSR_1	Initialized	—	—	—	—	—	—	—	Initiali
TCORA_0	Initialized	—	—	—	—	—	—	—	Initiali
TCORA_1	Initialized	—	—	—	—	—	—	—	Initiali

PWOERA	Initialized	—	—	—	—	—	—	—	—	Initialized
PWDRPB	Initialized	—	—	—	—	—	—	—	—	Initialized
PWDpra	Initialized	—	—	—	—	—	—	—	—	Initialized
PWSL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
PWDR0 to PWDR15	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
SMR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ICCR_0	Initialized	—	—	—	—	—	—	—	—	Initialized
BRR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ICSR_0	Initialized	—	—	—	—	—	—	—	—	Initialized
SCR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
TDR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
SSR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
RDR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
SCMR_0	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ICDR_0	—	—	—	—	—	—	—	—	—	—
SARX_0	Initialized	—	—	—	—	—	—	—	—	Initialized
ICMR_0	Initialized	—	—	—	—	—	—	—	—	Initialized
SAR_0	Initialized	—	—	—	—	—	—	—	—	Initialized
ADDRAH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRAL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRBH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRBL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRCH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRCL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRDH	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADDRDL	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized
ADCSR	Initialized	—	Initialized	—	Initialized	Initialized	Initialized	Initialized	Initialized	Initialized

TCR_X	Initialized	—	—	—	—	—	—	—	Initial
TCR_Y	Initialized	—	—	—	—	—	—	—	Initial
KMIMR	Initialized	—	—	—	—	—	—	—	Initial
TCSR_X	Initialized	—	—	—	—	—	—	—	Initial
TCSR_Y	Initialized	—	—	—	—	—	—	—	Initial
KMPCR	Initialized	—	—	—	—	—	—	—	Initial
TICRR	Initialized	—	—	—	—	—	—	—	Initial
TCORA_Y	Initialized	—	—	—	—	—	—	—	Initial
KMIMRA	Initialized	—	—	—	—	—	—	—	Initial
TICRF	Initialized	—	—	—	—	—	—	—	Initial
TCORB_Y	Initialized	—	—	—	—	—	—	—	Initial
IDR_1	—	—	—	—	—	—	—	—	—
TCNT_X	Initialized	—	—	—	—	—	—	—	Initial
TCNT_Y	Initialized	—	—	—	—	—	—	—	Initial
ODR_1	—	—	—	—	—	—	—	—	—
TCORC	Initialized	—	—	—	—	—	—	—	Initial
TISR	Initialized	—	—	—	—	—	—	—	Initial
STR_1	Initialized	—	—	—	—	—	—	—	Initial
TCORA_X	Initialized	—	—	—	—	—	—	—	Initial
TCORB_X	Initialized	—	—	—	—	—	—	—	Initial
DADR0	Initialized	—	—	—	—	—	—	—	Initial
DADR1	Initialized	—	—	—	—	—	—	—	Initial
DACR	Initialized	—	—	—	—	—	—	—	Initial
IDR_2	—	—	—	—	—	—	—	—	—
TCONR1	Initialized	—	—	—	—	—	—	—	Initial
ODR_2	—	—	—	—	—	—	—	—	—



- Notes:
1. Can be used on the H8S/2160B and H8S/2161B.
 2. Not supported by the H8S/2148B and H8S/2145B (5-V version).

H'FE1C	PCNOCR		
H'FE1D	PDNOCR		
H'FE20	TWR0MW	MSTP = 0, (HI12E = 0)*	MSTP = 0, (HI12E = 0)*
	TWR0SW		
H'FE21	TWR1		
H'FE22	TWR2		
H'FE23	TWR3		
H'FE24	TWR4		
H'FE25	TWR5		
H'FE26	TWR6		
H'FE27	TWR7		
H'FE28	TWR8		
H'FE29	TWR9		
H'FE2A	TWR10		
H'FE2B	TWR11		
H'FE2C	TWR12		
H'FE2D	TWR13		
H'FE2E	TWR14		
H'FE2F	TWR15		

H'FE36	SIRQCR0			
H'FE37	SIRQCR1			
H'FE38	IDR1			
H'FE39	ODR1			
H'FE3A	STR1			
H'FE3C	IDR2			
H'FE3D	ODR2			
H'FE3E	STR2			
H'FE3F	HISEL			
H'FE40	HICR0			
H'FE41	HICR1			
H'FE42	HICR2			
H'FE43	HICR3			
H'FE44	WUEMRB	No condition	No condition	IN
H'FE46	PGODR	—	No condition	PC
H'FE47	PGPIN (read)			
	PGDDR (write)			
H'FE48	PEODR			
H'FE49	PFODR			
H'FE4A	PEPIN (read)			
	PEDDR (write)			
H'FE4B	PFPIN (read)			
	PFDDR (write)			
H'FE4C	PCODR			
H'FE4D	PDODR			

H'FE81	IDR_3			
H'FE82	ODR_3			
H'FE83	STR_3			
H'FE84	IDR_4			
H'FE85	ODR_4			
H'FE86	STR_4			
H'FED4	ICXR_0	No condition	No condition	II
H'FED5	ICXR_1			II
H'FED8	KBCRH_0	MSTP2 = 0	MSTP2 = 0	K
H'FED9	KBCRL_0			b
H'FEDA	KBBR_0			c
H'FEDC	KBCRH_1			
H'FEDD	KBCRL_1			
H'FEDE	KBBR_1			
H'FEE0	KBCRH_2			
H'FEE1	KBCRL_2			
H'FEE2	KBBR_2			
H'FEE4	KBCOMP	No condition	No condition	II
H'FEE6	DDCSWR	MSTP4 = 0	MSTP4 = 0	II
H'FEE8	ICRA	No condition	No condition	II
H'FEE9	ICRB			
H'FEEA	ICRC			
H'FEEB	ISR			
H'FEEC	ISCRH			
H'FEED	ISCR_L			

H'FEF3	DTVECR			
H'FEF4	ABRKCR	No condition	No condition	IN
H'FEF5	BARA			
H'FEF6	BARB			
H'FEF7	BARC			
H'FF80	FLMCR1			
H'FF81	FLMCR2			
H'FF82	PCSR	FLSHE = 0 in STCR	FLSHE = 0 in STCR	PV
	EBR1	FLSHE = 1 in STCR	FLSHE = 1 in STCR	FL
H'FF83	SYSCR2	FLSHE = 0 in STCR	FLSHE = 0 in STCR	SY
	EBR2	FLSHE = 1 in STCR	FLSHE = 1 in STCR	FL
H'FF84	SBYCR	FLSHE = 0 in STCR	FLSHE = 0 in STCR	SY
H'FF85	LPWRCR			
H'FF86	MSTPCRH			
H'FF87	MSTPCRL			
H'FF88	SMR_1			
	ICCR_1	MSTP3 = 0, IICE = 1 in STCR	MSTP3 = 0, IICE = 1 in STCR	IIC
H'FF89	BRR_1	MSTP6 = 0, IICE = 0 in STCR	MSTP6 = 0, IICE = 0 in STCR	SC
	ICSR_1	MSTP3 = 0, IICE = 1 in STCR	MSTP3 = 0, IICE = 1 in STCR	IIC

		STCR		STCR	
	ICDR_1	MSTP3 = 0, IICE = 1 in STCR	ICE = 1 in ICCR1	MSTP3 = 0, IICE = 1 in STCR	ICE = 1 in ICCR1
	SARX_1		ICE = 0 in ICCR1		ICE = 0 in ICCR1
H'FF8F	ICMR_1		ICE = 1 in ICCR1		ICE = 1 in ICCR1
	SAR_1		ICE = 0 in ICCR1		ICE = 0 in ICCR1
H'FF90	TIER	MSTP13 = 0		MSTP13 = 0	
H'FF91	TCSR				
H'FF92	FRCH				
H'FF93	FRCL				
H'FF94	OCRAH	MSTP13 = 0	OCRS = 0 in TOCR	MSTP13 = 0	OCRS = 0 in TOCR
	OCRBH		OCRS = 1 in TOCR		OCRS = 1 in TOCR
H'FF95	OCRAL		OCRS = 0 in TOCR		OCRS = 0 in TOCR
	OCRBL		OCRS = 1 in TOCR		OCRS = 1 in TOCR
H'FF96	TCR				
H'FF97	TOCR				
H'FF98	ICRAH		ICRS = 0 in TOCR		ICRS = 0 in TOCR
	OCRARH		ICRS = 1 in TOCR		ICRS = 1 in TOCR

	OCRAFH		ICRS = 1 in TOCR		ICRS = 1 in TOCR
H'FF9B	ICRBL		ICRS = 0 in TOCR		ICRS = 0 in TOCR
	OCRAFL		ICRS = 1 in TOCR		ICRS = 1 in TOCR
H'FF9C	ICRCH		ICRS = 0 in TOCR		ICRS = 0 in TOCR
	OCRDMH		ICRS = 1 in TOCR		ICRS = 1 in TOCR
H'FF9D	ICRCL		ICRS = 0 in TOCR		ICRS = 0 in TOCR
	OCRDML		ICRS = 1 in TOCR		ICRS = 1 in TOCR
H'FF9E	ICRDH				
H'FF9F	ICRDL				
H'FFA0	SMR_2	MSTP5 = 0, IICE = 0 in STCR		MSTP5 = 0, IICE = 0 in STCR	SC
	DADRAH	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/DADRB	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/DADRB
	DACR		REGS = 1 in DACNT/DADRB		REGS = 1 in DACNT/DADRB
H'FFA1	BRR_2	MSTP5 = 0, IICE = 0 in STCR		MSTP5 = 0, IICE = 0 in STCR	SC
	DADRAL	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/DADRB	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/DADRB

		STCR	STCR	
	DADR BH	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/ DADRB	MSTP11 = 0, IICE = 1 in STCR
	DACNTH		REGS = 1 in DACNT/ DADRB	
H'FFA7	DADRBL		REGS = 0 in DACNT/ DADRB	
	DACNTL	REGS = 1 in DACNT/ DADRB	REGS = 0 in DACNT/ DADRB	REGS = 1 in DACNT/ DADRB
H'FFA8	TCSR_0	No condition	No condition	V
	TCNT_0 (write)			
H'FFA9	TCNT_0 (read)			
H'FFAA	PAODR	No condition	No condition	P
H'FFAB	PAPIN (read)			
	PADDR (write)			
H'FFAC	P1PCR			
H'FFAD	P2PCR			
H'FFAE	P3PCR			
H'FFB0	P1DDR			
H'FFB1	P2DDR			
H'FFB2	P1DR			
H'FFB3	P2DR			
H'FFB4	P3DDR			
H'FFB5	P4DDR			

H'FFBB	P6DR			
H'FFBC	PBODR			
H'FFBD	P8DDR (write)			
	PBPIN (read)			
H'FFBE	P7PIN (read)			
	PBDDR (write)			
H'FFBF	P8DR			
H'FFC0	P9DDR			
H'FFC1	P9DR			
H'FFC2	IER	No condition	No condition	IN
H'FFC3	STCR	No condition	No condition	SY
H'FFC4	SYSCR			
H'FFC5	MDCR			
H'FFC6	BCR	No condition	No condition	BS
H'FFC7	WSCR			
H'FFC8	TCR_0	MSTP12 = 0	MSTP12 = 0	TM
H'FFC9	TCR_1			
H'FFCA	TCSR_0			
H'FFCB	TCSR_1			
H'FFCC	TCORA_0			
H'FFCD	TCORA_1			
H'FFCE	TCORB_0			
H'FFCF	TCORB_1			
H'FFD0	TCNT_0			
H'FFD1	TCNT_1			

H'FFD7	PWDR0 to PWDR15				
H'FFD8	SMR_0	MSTP7 = 0, IICE = 0 in STCR		MSTP7 = 0, IICE = 0 in STCR	
	ICCR_0	MSTP4 = 0, IICE = 1 in STCR		MSTP4 = 0, IICE = 1 in STCR	
H'FFD9	BRR_0	MSTP7 = 0, IICE = 0 in STCR		MSTP7 = 0, IICE = 0 in STCR	
	ICSR_0	MSTP4 = 0, IICE = 1 in STCR		MSTP4 = 0, IICE = 1 in STCR	
H'FFDA	SCR_0	MSTP7 = 0		MSTP7 = 0	
H'FFDB	TDR_0				
H'FFDC	SSR_0				
H'FFDD	RDR_0				
H'FFDE	SCMR_0	MSTP7 = 0, IICE = 0 in STCR		MSTP7 = 0, IICE = 0 in STCR	
	ICDR_0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0
	SARX_0		ICE = 0 in ICCR0		ICE = 0 in ICCR0
H'FFDF	ICMR_0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0
	SAR_0		ICE = 0 in ICCR0		ICE = 0 in ICCR0

H'FFE5	ADDRCL				
H'FFE6	ADDRDH				
H'FFE7	ADDRDL				
H'FFE8	ADCSR				
H'FFE9	ADCR				
H'FFEA	TCSR_1	No condition		No condition	
	TCNT_1 (write)				
H'FFEB	TCNT_1 (read)				
H'FFF0	HICR	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCR_X	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
	TCR_Y		TMRX/Y = 1 in TCONRS		TMRX/Y = 1 in TCONRS
H'FFF1	KMIMR	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCSR_X	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
	TCSR_Y		TMRX/Y = 1 in TCONRS		TMRX/Y = 1 in TCONRS

H'FFF3	KMIMRA	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TICRF	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
	TCORB_Y		TMRX/Y = 1 in TCONRS		TMRX/Y = 1 in TCONRS
H'FFF4	IDR_1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCNT_X	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
	TCNT_Y		TMRX/Y = 1 in TCONRS		TMRX/Y = 1 in TCONRS
H'FFF5	ODR_1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCORC	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
	TISR		TMRX/Y = 1 in TCONRS		TMRX/Y = 1 in TCONRS
H'FFF6	STR_1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCORA_X	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS
H'FFF7	TCORB_X				

	TCONRI	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR	Tin co
H'FFFD	ODR_2	MSTP2 = 0, HIE = 1 in SYSCR	MSTP2 = 0, HIE = 1 in SYSCR	XB
	TCONRO	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR	Tin co
H'FFFE	STR_2	MSTP2 = 0, HIE = 1 in SYSCR	MSTP2 = 0, HIE = 1 in SYSCR	XB
	TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR	Tin co
H'FFFF	SEDGR			

Note: * Although setting the XBS corresponding bits does not affected to the LPC of the HI12E bit in SYSCR2 must not be set to 1 to use the LPC according to the depending on the program development tool (emulator) configuration.

Table 27.1 lists the absolute maximum ratings.

Table 27.1 Absolute Maximum Ratings

Item	Symbol	Value
Power supply voltage	V_{CC}, V_{CL}	-0.3 to +4.3
I/O buffer power supply voltage	V_{CCB}	-0.3 to +7.0
Input voltage (except ports 6, 7, and A, P97, P86, P52, and P42) (Ports C to G are added in the H8S/2160B and H8S/2161B.)	V_{in}	-0.3 to $V_{CC} + 0.3$
Input voltage (CIN input not selected for port 6)	V_{in}	-0.3 to $V_{CC} + 0.3$
Input voltage (CIN input not selected for port A)	V_{in}	-0.3 to $V_{CCB} + 0.3$
Input voltage (CIN input selected for port 6)	V_{in}	-0.3 V to lower of voltage $V_{CC} + 0.3$ and $AV_{CC} + 0.3$
Input voltage (CIN input selected for port A)	V_{in}	-0.3 V to lower of voltage $V_{CCB} + 0.3$ and $AV_{CC} + 0.3$
Input voltage (P97, P86, P52, P42)	V_{in}	-0.3 to +7.0
Input voltage (port 7)	V_{in}	-0.3 to $AV_{CC} + 0.3$
Reference supply voltage	AV_{ref}	-0.3 to $AV_{CC} + 0.3$
Analog power supply voltage	AV_{CC}	-0.3 to +4.3
Analog input voltage	V_{AN}	-0.3 to $AV_{CC} + 0.3$
Operating temperature	T_{opr}	-20 to +75
Operating temperature (flash memory programming/erasing)	T_{opr}	-20 to +75
Storage temperature	T_{stg}	-55 to +125

Caution: Permanent damage to the chip may result if absolute maximum ratings are exceeded. Ensure so that the impressed voltage does not exceed 4.3 V for pins for which the maximum rating is determined by the voltage on the V_{CC} , AV_{CC} , and V_{CL} pins, and -0.3 V for pins for which the maximum rating is determined by V_{CCB} . The V_{CC} and V_{CL} pins must be connected to the Vcc power supply.

Item		Symbol	Min	Typ	Max	Unit	
Schmitt trigger input voltage	P67 to P60 ^{*2 *6} , KIN15 to KIN8 ^{*7} , IRQ2 to IRQ0 ^{*3} , IRQ5 to IRQ3	(1) ^{*8}	V_T^-	$V_{CC} \times 0.2$	—	—	V
			V_T^+	$V_{CC}B \times 0.2$	—	—	
			V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$ $V_{CC}B \times 0.05$	—	—	
Schmitt trigger input voltage (in level switching) ^{*6}	P67 to P60 (KWUL = 00)		V_T^-	$V_{CC} \times 0.2$	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	
	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	
			V_T^+	—	—	$V_{CC} \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	
	P67 to P60 (KWUL = 10)		V_T^-	$V_{CC} \times 0.4$	—	—	
			V_T^+	—	—	$V_{CC} \times 0.8$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.03$	—	—	
	P67 to P60 (KWUL = 11)		V_T^-	$V_{CC} \times 0.45$	—	—	
			V_T^+	—	—	$V_{CC} \times 0.9$	
			$V_T^+ - V_T^-$	0.05	—	—	
Input high voltage	RES, STBY, NMI, MD1, MD0 EXTAL PA7 to PA0 ^{*7}	(2)	V_{IH}	$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	V
				$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	
				$V_{CC}B \times 0.7$	—	$V_{CC}B + 0.3$	

		H8S/2160B and H8S/2161B.)					
Input low voltage	RES, STBY, MD1, MD0	(3)	V_{IL}	-0.3	—	$V_{CC} \times 0.1$	V
	PA7 to PA0			-0.3	—	$V_{CC}B \times 0.2$	
						0.8	
	NMI, EXTAL, input pins except (1) and (3) above (Ports C to G are added in the H8S/2160B and H8S/2161B.)			-0.3	—	$V_{CC} \times 0.2$	
Output high voltage	All output pins (except P97, P86, P52, and P42) ^{*4 *5 *8} (Ports C to F are added in the H8S/2160B and H8S/2161B.)		V_{OH}	$V_{CC} - 0.5$ $V_{CC}B - 0.5$	—	—	V
				$V_{CC} - 1.0$ $V_{CC}B - 1.0$	—	—	V
	P97, P86, P52, and P42 ^{*4} (Port G is added in the H8S/2160B and H8S/2161B.)			0.5	—	—	V

Notes: 1. Do not leave the AV_{cc} , AV_{ref} , and AV_{ss} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 0 V to 3.6 V to AV_{cc} and AV_{ref} pins by connection to the power supply (V_{cc}), or by other method. Ensure that $AV_{ref} \leq AV_{cc}$.

2. P67 to P60 include peripheral module inputs multiplexed on those pins.

3. $\overline{IRQ2}$ includes the \overline{ADTRG} signal multiplexed on that pin.

4. P52/SCK0/SCL0, P97/SDA0, P86/SCK1/SCL1, P42/SCK2/SDA1, and port C include NMOS push-pull outputs.

When the SCL0, SDA0, SCL1, or SDA1 (ICE = 1) pin is used as an output, it is an open-drain output. Therefore, an external pull-up resistor must be connected to the output high level.

P52/SCK0, P97, P86/SCK1, P42/SCK2 (ICE = 0), and port G high levels are NMOS.

An external pull-up resistor is necessary to provide high-level output from SCL0 and SCK2.

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus driver is selected is determined separately.

6. The upper limit of the port 6 applied voltage is $V_{cc} + 0.3$ V when CIN input is selected, and the lower of $V_{cc} + 0.3$ V and $AV_{cc} + 0.3$ V when CIN input is selected. When a pin is in output mode, the output voltage is equivalent to the applied voltage.

7. The upper limit of the port A applied voltage is $V_{ccB} + 0.3$ V when CIN input is selected, and the lower of $V_{ccB} + 0.3$ V and $AV_{cc} + 0.3$ V when CIN input is selected. When a pin is in output mode, the output voltage is equivalent to the applied voltage.

8. The port A characteristics depend on V_{ccB} , and the other pins characteristics depend on V_{cc} .

9. For flash memory programming/erasure, the applicable range is $V_{cc} = 3.0$ V.

current	MD0		—	—	1.0	
	Port 7		—	—	1.0	
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A ^{*4} , and B (Ports C to G are added in the H8S/2160B and H8S/2161B.)	$ I_{TSL} $	—	—	1.0	μA
Input pull-up MOS current	Ports 1 to 3	$-I_p$	5	—	150	μA
	Ports 6 (P6PUE = 0) and B (Ports C to F are added in the H8S/2160B and H8S/2161B.)		30	—	300	
	Ports A ^{*4}		30	—	600	
	Port 6 (P6PUE = 1)		3	—	100	
Input capacitance	RES (4)	C_{in}	—	—	80	pF
	NMI		—	—	50	pF
	P52, P97, P42, P86, PA7 to PA2		—	—	20	pF
	Input pins except (4) above (Ports C to G are added in the H8S/2160B and H8S/2161B.)		—	—	15	pF

supply current	Idle		—	0.01	5.0	μA	A	to
Reference power supply current	During A/D conversion	I_{ref}	—	0.5	1.0	mA		
	During A/D, D/A conversion		—	2.0	5.0			
	Idle		—	0.01	5.0	μA	A	to
Analog power supply voltage*1	V_{CC}		2.7	—	3.6	V	C	
			2.0	—	3.6		l	
RAM standby voltage		V_{RAM}	2.0	—	—	V		

Notes: 1. Do not leave the AV_{CC} , AV_{ref} and AV_{SS} pins open even if the A/D converter and converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 0 V to 3.6 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or by connection to ground (GND) method. Ensure that $AV_{ref} \leq AV_{CC}$.

- Current dissipation values are for $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CC}B - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$ with all output pins unloaded and the on-chip pull-up MOSs in the high-impedance state.
- The values are for $V_{RAM} \leq V_{CC} < 2.7 \text{ V}$, $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CC}B - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$.
- The port A characteristics depend on $V_{CC}B$, and the other pins characteristics depend on V_{CC} .
- For flash memory programming/erasure, the applicable range is $V_{CC} = 3.0 \text{ V}$.

	PB1, PB0						
Input low voltage	P37 to P30, P83 to P80, PB1, PB0	V_{IL}	—	$V_{CC} \times 0.3$	V		
Output high voltage	P37, P33 to P30, P82 to P80, PB1, PB0	V_{OH}	$V_{CC} \times 0.9$	—	V	$I_{OH} = -$	
Output low voltage	P37, P33 to P30, P82 to P80, PB1, PB0	V_{OL}	—	$V_{CC} \times 0.1$	V	$I_{OL} = 1$	

Note: * Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in 2.0 V to 3.6 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}) or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

	Ports 1 to 3		—	—	2
	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1 to 3	ΣI_{OL}	—	—	40
	Total of all output pins, including the above		—	—	60
Permissible output high current (per pin)	All output pins	$-I_{OH}$	—	—	2
Permissible output high current (total)	Total of all output pins	$\Sigma -I_{OH}$	—	—	30

- Notes: 1. To protect chip reliability, do not exceed the output current values in table 27.1.
2. When driving a Darlington pair or LED, always insert a current-limiting resistor on the output line, as shown in figures 27.1 and 27.2.

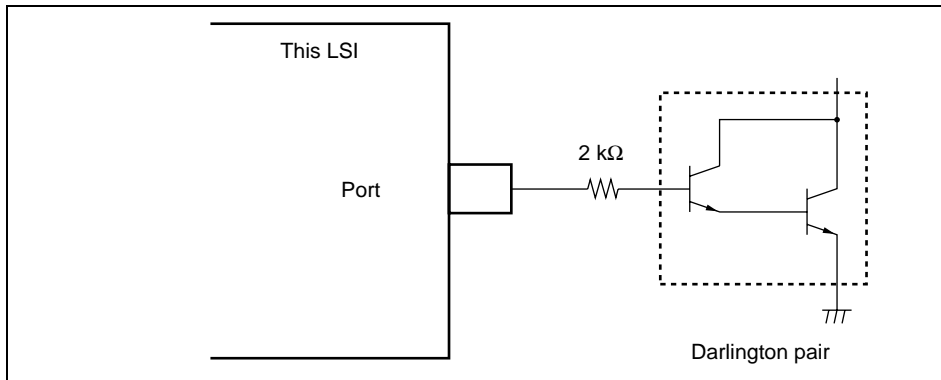


Figure 27.1 Darlington Pair Drive Circuit (Example)

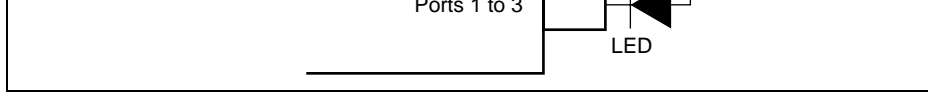


Figure 27.2 LED Drive Circuit (Example)

Table 27.4 Bus Drive Characteristics

Conditions: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -20\text{ to }+75^\circ\text{C}$

Applicable Pins: SCL1, SCL0, SDA1, SDA0 (bus drive function selected)

Item	Symbol	Min	Typ	Max	Unit	Test Con
Schmitt trigger input voltage	V_T^-	$V_{CC} \times 0.3$	—	—	V	$V_{CC} = 2.7\text{ V}$
	V_T^+	—	—	$V_{CC} \times 0.7$		$V_{CC} = 2.7\text{ V}$
	$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—		$V_{CC} = 2.7\text{ V}$
Input high voltage	V_{IH}	$V_{CC} \times 0.7$	—	5.5	V	$V_{CC} = 2.7\text{ V}$
Input low voltage	V_{IL}	-0.5	—	$V_{CC} \times 0.3$		$V_{CC} = 2.7\text{ V}$
Output low voltage	V_{OL}	—	—	0.5	V	$I_{OL} = 8\text{ mA}$
		—	—	0.4		$I_{OL} = 3\text{ mA}$
Input capacitance	C_{in}	—	—	20	pF	$V_{in} = 0\text{ V}$, $T_a = 25^\circ\text{C}$
Three-state leakage current (off state)	$ I_{TSL} $	—	—	1.0	μA	$V_{in} = 0.5\text{ to }1.5\text{ V}$
SCL, SDA output fall time	t_{of}	$20 + 0.1Cb$	—	250	ns	$V_{CC} = 2.7\text{ V}$

27.1.3 AC Characteristics

Figure 27.3 shows the test conditions for the AC characteristics.

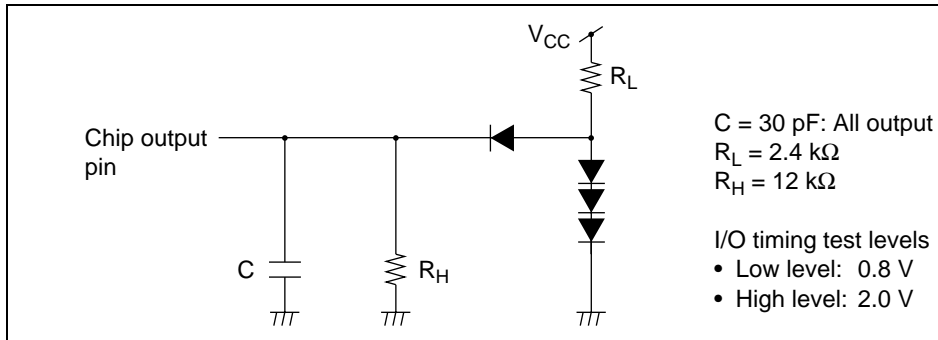


Figure 27.3 Output Load Circuit

Item	Symbol	Condition		Unit
		10 MHz		
		Min	Max	
Clock cycle time	t_{cyc}	100	500	ns
Clock high pulse width	t_{CH}	30	—	ns
Clock low pulse width	t_{CL}	30	—	ns
Clock rise time	t_{Cr}	—	20	ns
Clock fall time	t_{Cf}	—	20	ns
Oscillation settling time at reset (crystal)	t_{OSC1}	20	—	ms
Oscillation settling time in software standby (crystal)	t_{OSC2}	8	—	ms
External clock output stabilization delay time	t_{DEXT}	500	—	μs

Item	Symbol	10 MHz		Unit	T C
		Min	Max		
$\overline{\text{RES}}$ setup time	t_{RESS}	300	—	ns	F
$\overline{\text{RES}}$ pulse width	t_{RESW}	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	250	—	ns	F
NMI hold time (NMI)	t_{NMIH}	10	—	ns	
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	ns	
IRQ setup time ($\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$)	t_{IRQS}	250	—	ns	
IRQ hold time($\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$)	t_{IRQH}	10	—	ns	
IRQ pulse width ($\overline{\text{IRQ7}}$, $\overline{\text{IRQ6}}$, $\overline{\text{IRQ2}}$ to $\overline{\text{IRQ0}}$) (exiting software standby mode)	t_{IRQW}	200	—	ns	

Item	Symbol	Min	Max	Unit	Te
Address delay time	t_{AD}	—	40	ns	Fi
Address setup time	t_{AS}	$0.5 \times t_{cyc} - 30$	—	ns	to
Address hold time	t_{AH}	$0.5 \times t_{cyc} - 20$	—	ns	
\overline{CS} delay time (\overline{IOS})	t_{CSD}	—	40	ns	
\overline{AS} delay time	t_{ASD}	—	60	ns	
\overline{RD} delay time 1	t_{RSD1}	—	60	ns	
\overline{RD} delay time 2	t_{RSD2}	—	60	ns	
Read data setup time	t_{RDS}	35	—	ns	
Read data hold time	t_{RDH}	0	—	ns	
Read data access time 1	t_{ACC1}	—	$1.0 \times t_{cyc} - 60$	ns	
Read data access time 2	t_{ACC2}	—	$1.5 \times t_{cyc} - 50$	ns	
Read data access time 3	t_{ACC3}	—	$2.0 \times t_{cyc} - 60$	ns	
Read data access time 4	t_{ACC4}	—	$2.5 \times t_{cyc} - 50$	ns	
Read data access time 5	t_{ACC5}	—	$3.0 \times t_{cyc} - 60$	ns	
\overline{HWR} , \overline{LWR} delay time 1	t_{WRD1}	—	60	ns	
\overline{HWR} , \overline{LWR} delay time 2	t_{WRD2}	—	60	ns	
\overline{HWR} , \overline{LWR} pulse width 1	t_{WSW1}	$1.0 \times t_{cyc} - 40$	—	ns	
\overline{HWR} , \overline{LWR} pulse width 2	t_{WSW2}	$1.5 \times t_{cyc} - 40$	—	ns	
Write data delay time	t_{WDD}	—	60	ns	
Write data setup time	t_{WDS}	0	—	ns	
Write data hold time	t_{WDH}	20	—	ns	
\overline{WAIT} setup time	t_{WTS}	60	—	ns	
\overline{WAIT} hold time	t_{WTH}	10	—	ns	

Address delay time	t_{AD}	—	60	ns
Address setup time	t_{AS}	$0.5 \times t_{cyc} - 30$	—	ns
Address hold time	t_{AH}	$0.5 \times t_{cyc} - 20$	—	ns
\overline{CS} delay time (\overline{IOS})	t_{CSD}	—	60	ns
\overline{AS} delay time	t_{ASD}	—	60	ns
\overline{RD} delay time 1	t_{RSD1}	—	60	ns
\overline{RD} delay time 2	t_{RSD2}	—	60	ns
Read data setup time	t_{RDS}	35	—	ns
Read data hold time	t_{RDH}	0	—	ns
Read data access time 1	t_{ACC1}	—	$1.0 \times t_{cyc} - 80$	ns
Read data access time 2	t_{ACC2}	—	$1.5 \times t_{cyc} - 50$	ns
Read data access time 3	t_{ACC3}	—	$2.0 \times t_{cyc} - 80$	ns
Read data access time 4	t_{ACC4}	—	$2.5 \times t_{cyc} - 50$	ns
Read data access time 5	t_{ACC5}	—	$3.0 \times t_{cyc} - 80$	ns
\overline{HWR} , \overline{LWR} delay time 1	t_{WRD1}	—	60	ns
\overline{HWR} , \overline{LWR} delay time 2	t_{WRD2}	—	60	ns
\overline{HWR} , \overline{LWR} pulse width 1	t_{WSW1}	$1.0 \times t_{cyc} - 40$	—	ns
\overline{HWR} , \overline{LWR} pulse width 2	t_{WSW2}	$1.5 \times t_{cyc} - 40$	—	ns
Write data delay time	t_{WDD}	—	60	ns
Write data setup time	t_{WDS}	0	—	ns
Write data hold time	t_{WDH}	20	—	ns
\overline{WAIT} setup time	t_{WTS}	60	—	ns
\overline{WAIT} hold time	t_{WTH}	10	—	ns

Item		Symbol	Condition		Unit	Test		
			10 MHz					
			Min	Max				
I/O ports	Output data delay time	t_{PWD}	—	100	ns	Figure		
	Input data setup time	t_{PRS}	50	—				
	Input data hold time	t_{PRH}	50	—				
FRT	Timer output delay time	t_{FTOD}	—	100	ns	Figure		
	Timer input setup time	t_{FTIS}	50	—				
	Timer clock input setup time	t_{FTCS}	50	—				
	Timer clock pulse width						t_{cyc}	
	Single edge	t_{FTCWH}	1.5	—				
	Both edges	t_{FTCWL}	2.5	—				
TMR	Timer output delay time	t_{TMOD}	—	100	ns	Figure		
	Timer reset input setup time	t_{TMRS}	50	—				
	Timer clock input setup time	t_{TMCS}	50	—				
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5			—	t_{cyc}
		Both edges	t_{TMCWL}	2.5			—	
PWM, PWMX	Pulse output delay time	t_{PWOD}	—	100	ns	Figure		
SCI	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{cyc}	Figure	
		Synchronous		6	—			
	Input clock pulse width	t_{SCKW}	0.4	0.6	t_{cyc}			
	Input clock rise time	t_{SCKr}	—	1.5	t_{cyc}			
	Input clock fall time	t_{SCKf}	—	1.5				

A/D converter	Trigger input setup time	t_{TRGS}	50	—	ns	Figure
WDT	\overline{RESO} output delay time	t_{RESD}	—	200	ns	Figure
	\overline{RESO} output pulse width	t_{RESOW}	132	—	t_{cyc}	

Note: * Only peripheral modules that can be used in subclock operation

Table 27.8 Timing of On-Chip Peripheral Modules (2)

Condition: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{CCB} = 2.7\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to max}$
operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$

Item	Symbol	Condition		Unit	Test		
		10 MHz					
		Min	Max				
XBS read cycle	$\overline{CS}/HA0$ setup time	t_{HAR}	10	—	ns	Figure	
	$\overline{CS}/HA0$ hold time	t_{HRA}	10	—	ns		
	\overline{IOR} pulse width	t_{HRPW}	220	—	ns		
	HDB delay time	t_{HRD}	—	200	ns		
	HDB hold time	t_{HRF}	0	40	ns		
	HIRQ delay time	t_{HIRQ}	—	200	ns		
XBS write cycle	$\overline{CS}/HA0$ setup time	t_{HAW}	10	—	ns		
	$\overline{CS}/HA0$ hold time	t_{HWA}	10	—	ns		
	\overline{IOW} pulse width	t_{HWPW}	100	—	ns		
	HDB setup time	Fast A20 gate not used	t_{HDW}	50	—	ns	
		Fast A20 gate used		85	—	ns	
	HDB hold time	t_{HWD}	25	—	ns		
	GA20 delay time	t_{HGA}	—	180	ns		

KCLK, KD input data hold time	t_{KBIS}	150	—	—	ns
KCLK, KD output delay time	t_{KBOD}	—	—	450	ns
KCLK, KD capacitive load	C_b	—	—	400	pF

Table 27.10 I²C Bus Timing

Conditions: $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{SS} = 0 \text{ V}$, $\phi = 5 \text{ MHz}$ to maximum operating frequency
 $T_a = -20 \text{ to } +75^\circ\text{C}$

Item	Symbol	Ratings			Unit	Test Conditions
		Min	Typ	Max		
SCL input cycle time	t_{SCL}	12	—	—	t_{cyc}	
SCL input high pulse width	t_{SCLH}	3	—	—	t_{cyc}	
SCL input low pulse width	t_{SCLL}	5	—	—	t_{cyc}	
SCL, SDA input rise time	t_{Sr}	—	—	7.5*	t_{cyc}	
SCL, SDA input fall time	t_{Sf}	—	—	300	ns	
SCL, SDA input spike pulse elimination time	t_{SP}	—	—	1	t_{cyc}	
SDA input bus free time	t_{BUF}	5	—	—	t_{cyc}	
Start condition input hold time	t_{STAH}	3	—	—	t_{cyc}	
Retransmission start condition input setup time	t_{STAS}	3	—	—	t_{cyc}	
Stop condition input setup time	t_{STOS}	3	—	—	t_{cyc}	
Data input setup time	t_{SDAS}	0.5	—	—	t_{cyc}	
Data input hold time	t_{SDAH}	0	—	—	ns	
SCL, SDA capacitive load	C_b	—	—	400	pF	

Note: * $17.5 t_{cyc}$ can be set according to the clock selected for use by the I²C module details, see section 16.6, Usage Notes.

Input clock pulse width (L)	t_{LCKL}	11	—	—
Transmit signal delay time	t_{TXD}	2	—	11
Transmit signal floating delay time	t_{OFF}	—	—	28
Receive signal setup time	t_{RXS}	7	—	—
Receive signal hold time	t_{RXH}	0	—	—

Item	Condition		
	10 MHz		
	Min	Typ	Max
Resolution	10	10	10
Conversion time	—	—	13.4
Analog input capacitance	—	—	20
Permissible signal-source impedance	—	—	5
Nonlinearity error	—	—	±7.0
Offset error	—	—	±7.5
Full-scale error	—	—	±7.5
Quantization error	—	—	±0.5
Absolute accuracy	—	—	±8.0

Item	Min	Typ	Max
Resolution	10	10	10
Conversion time	—	—	13.4
Analog input capacitance	—	—	20
Permissible signal-source impedance	—	—	5
Nonlinearity error	—	—	±11.0
Offset error	—	—	±11.5
Full-scale error	—	—	±11.5
Quantization error	—	—	±0.5
Absolute accuracy	—	—	±12.0

		Condition		
		10 MHz		
Item		Min	Typ	Max
Resolution		8	8	8
Conversion time	With 20 pF load capacitance	—	—	10
Absolute accuracy	With 2 M Ω load resistance	—	± 2.0	± 3.0
	With 4 M Ω load resistance	—	—	± 2.0

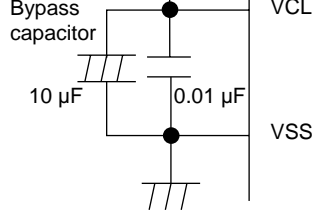
Programming time ^{*1 *2 *4}	t_p	—	10	200	ms/ 128 bytes	
Erase time ^{*1 *3 *6}	t_E	—	100	1200	ms/block	
Reprogramming count	N_{WEC}	100 ^{*8}	10,000 ^{*9}	—	times	
Data retention time ^{*10}	t_{DRP}	10	—	—	Years	
Programming	Wait time after SWE-bit setting ^{*1}	x	1	—	—	μ s
	Wait time after PSU-bit setting ^{*1}	y	50	—	—	μ s
	Wait time after P-bit setting ^{*1 *4}	z1	28	30	32	μ s
		z2	198	200	202	μ s
		z3	8	10	12	μ s
	Wait time after P-bit clear ^{*1}	α	5	—	—	μ s
	Wait time after PSU-bit clear ^{*1}	β	5	—	—	μ s
	Wait time after PV-bit setting ^{*1}	γ	4	—	—	μ s
	Wait time after dummy write ^{*1}	ϵ	2	—	—	μ s
	Wait time after PV-bit clear ^{*1}	η	2	—	—	μ s
	Wait time after SWE-bit clear ^{*1}	θ	100	—	—	μ s
	Maximum programming count ^{*1 *4 *5}	N	—	—	1000	times
	Erase	Wait time after SWE-bit setting ^{*1}	x	1	—	—
Wait time after ESU-bit setting ^{*1}		y	100	—	—	μ s
Wait time after E-bit setting ^{*1 *6}		z	10	—	100	ms
Wait time after E-bit clear ^{*1}		α	10	—	—	μ s
Wait time after ESU-bit clear ^{*1}		β	10	—	—	μ s
Wait time after EV-bit setting ^{*1}		γ	20	—	—	μ s
Wait time after dummy write ^{*1}		ϵ	2	—	—	μ s
Wait time after EV-bit clear ^{*1}		η	4	—	—	μ s
Wait time after SWE-bit clear ^{*1}		θ	100	—	—	μ s
Maximum erase count ^{*1 *6 *7}	N	—	—	120	times	

z1, z2 and z3 to allow programming within the maximum programming time.
The wait time after P-bit setting (z1, z2, and z3) should be alternated according to the number of writes (n) as follows:

$1 \leq n \leq 6$ z1 = 30 μ s, z3 = 10 μ s

$7 \leq n \leq 1000$ z2 = 200 μ s

6. Maximum erase time (t_E (max))
 t_E (max) = Wait time after E-bit setting (z) \times maximum erase count (N)
7. The maximum number of erases (N) should be set according to the actual erase time to allow erasing within the maximum erase time (t_E (max)).
8. Minimum number of times for which all characteristics are guaranteed after erasing (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally function within the specified range value).
10. Data retention characteristic when rewriting is performed within the specified range including the minimum value.



< Vcc = 2.7 V to 3.6 V >

Connect the Vcc power supply to the chip's VCL pin in the same way as the VCC pins.

It is recommended that a bypass capacitor be connected to the power supply pins. (Values are reference values.)

Figure 27.4 Connection of VCL Capacitor

Power supply voltage ^{*1}	V_{CC}	-0.3 to +7.0
I/O buffer power supply voltage (power supply for port A)	V_{CCB}	-0.3 to +7.0
Power supply voltage (3-V version product) ^{*1}	V_{CC}	-0.3 to +4.3
Power supply voltage (VCL pin) ^{*2}	V_{CL}	-0.3 to +4.3
Input voltage (except ports 6, 7, and A, P97, P86, P52, P42)	V_{in}	-0.3 to $V_{CC} + 0.3$
Input voltage (CIN input not selected for port 6)	V_{in}	-0.3 to $V_{CC} + 0.3$
Input voltage (CIN input not selected for port A)	V_{in}	-0.3 to $V_{CCB} + 0.3$
Input voltage (CIN input selected for port 6)	V_{in}	-0.3 V to lower of voltages $V_{CC} + 0.3$ and $AV_{CC} + 0.3$
Input voltage (CIN input selected for port A)	V_{in}	-0.3 V to lower of voltages $V_{CCB} + 0.3$ and $AV_{CC} + 0.3$
Input voltage (P97, P86, P52, P42)	V_{in}	-0.3 to +7.0
Input voltage (port 7)	V_{in}	-0.3 to $AV_{CC} + 0.3$
Reference supply voltage	AV_{ref}	-0.3 to $AV_{CC} + 0.3$
Analog power supply voltage	AV_{CC}	-0.3 to +7.0
Analog power supply voltage (3-V version product)	AV_{CC}	-0.3 to +4.3
Analog input voltage	V_{AN}	-0.3 to $AV_{CC} + 0.3$
Operating temperature	T_{opr}	Normal specification product: -20 to + Wide range temperature specification product: -40 to +85

- Notes:
1. Voltage applied to the VCC1 pin. Since both the VCC1 pin and VCL pin are connected to the VCC power supply on low-power voltage (3-V) products, VCL ratings should not be exceeded.
 2. Power supply voltage pin used for operation within the chip. Do not apply power supply voltage to the VCL pin on 5-V/4-V products. Be sure to insert an external capacitor between the VCL pin and GND to regulate the internal voltage.

Item		Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60 (KWUL = 00) ^{*2 *6} , KIN15 to KIN8 ^{*7 *8} , IRQ2 to IRQ0 ^{*3} , IRQ5 to IRQ3	(1) V _T ⁻	1.0	—	—	V
		V _T ⁺	—	—	V _{CC} × 0.7 V _{CC} B × 0.7	
		V _T ⁺ - V _T ⁻	0.4	—	—	
		<hr/>				
Schmitt trigger input voltage (in level switching) ^{*6}	P67 to P60 (KWUL = 01)	V _T ⁻	V _{CC} × 0.3	—	—	V
		V _T ⁺	—	—	V _{CC} × 0.7	
	V _T ⁺ - V _T ⁻	V _{CC} × 0.05	—	—		
	P67 to P60 (KWUL = 10)	V _T ⁻	V _{CC} × 0.4	—	—	
		V _T ⁺	—	—	V _{CC} × 0.8	
	V _T ⁺ - V _T ⁻	V _{CC} × 0.03	—	—		
	P67 to P60 (KWUL = 11)	V _T ⁻	V _{CC} × 0.45	—	—	
		V _T ⁺	—	—	V _{CC} × 0.9	
	V _T ⁺ - V _T ⁻	0.05	—	—		
	Input high voltage	RES, STBY, NMI, MD1, MD0 <hr/> EXTAL <hr/> PA7 to PA0 ^{*7} <hr/> Port 7 <hr/> P97, P86, P52, P42 <hr/> Input pins except (1) and (2) above	(2) V _{IH}	V _{CC} - 0.7	—	V _{CC} + 0.3
V _{CC} × 0.7			—	V _{CC} + 0.3		
V _{CC} B × 0.7			—	V _{CC} B + 0.3		
2.0			—	AV _{CC} + 0.3		
V _{CC} × 0.7			—	5.5		
2.0			—	V _{CC} + 0.3		

Output high voltage	All output pins (except P97, P86, P52, and P42)*5 *8	V_{OH}	$V_{CC} - 0.5$	—	—	V	I_{OH}
	P97, P86, P52, and P42*4		$V_{CCB} - 0.5$	—	—	V	I_{OH}
Output low voltage	All output pins (except RESO)*5	V_{OL}	—	—	0.4	V	I_{OL}
	Ports 1 to 3		—	—	1.0	V	I_{OL}
	RESO		—	—	0.4	V	I_{OL}

Notes: 1. Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 2.0 V to 5.5 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

- P67 to P60 include peripheral module inputs multiplexed on those pins.
- $\overline{IRQ2}$ includes the \overline{ADTRG} signal multiplexed on that pin.
- P52/SCK0/SCL0, P97/SDA0, P86/SCK1/SCL1, P42/SCK2/SDA1, and port C include NMOS push-pull outputs.

When the SCL0, SDA0, SCL1, or SDA1 (ICE = 1) pin is used as an output, it functions as an open-drain output. Therefore, an external pull-up resistor must be connected to the output high level.

P52/SCK0, P97, P86/SCK1, P42/SCK2 (ICE = 0), and port G high levels are determined by NMOS.

When the SCK0, SCK1, or SCK2 pin is used as an output, an external pull-up resistor must be connected in order to output high level.

- When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus driver is selected is determined separately.
- The upper limit of the port 6 applied voltage is $V_{CC} + 0.3$ V when CIN input is selected, and the lower of $V_{CC} + 0.3$ V and $AV_{CC} + 0.3$ V when CIN input is selected. When a pin is in output mode, the output voltage is equivalent to the applied voltage.
- The upper limit of the port A applied voltage is $V_{CCB} + 0.3$ V when CIN input is selected, and the lower of $V_{CCB} + 0.3$ V and $AV_{CC} + 0.3$ V when CIN input is selected. When a pin is in output mode, the output voltage is equivalent to the applied voltage.

	Item	Symbol	Min	Typ	Max	Unit
Input leakage current	$\overline{\text{RES}}$	$ I_{in} $	—	—	10.0	μA
	$\overline{\text{STBY}}$, NMI, MD1, MD0		—	—	1.0	
	Port 7		—	—	1.0	
Three-state leakage current (off state)	Ports 1 to 6 Ports 8, 9, A* ⁴ , B	$ I_{TSL} $	—	—	1.0	μA
Input pull-up MOS current	Ports 1 to 3	$-I_p$	30	—	300	μA
	Ports A* ⁴ , B, 6 (P6PUE = 0)		60	—	600	μA
	Port 6 (P6PUE = 1)		15	—	200	μA
Input capacitance	$\overline{\text{RES}}$ (4)	C_{in}	—	—	80	pF
	NMI		—	—	50	
	P52, P97, P42, P86, PA7 to PA2		—	—	20	
	Input pins except (4) above		—	—	15	
Current dissipation* ²	Normal operation	I_{CC}	—	55	70	mA
	Sleep mode		—	36	55	mA
	Standby mode* ³		—	1.0	5.0	μA
			—	—	20.0	
Analog power supply current	During A/D, D/A conversion	$A I_{CC}$	—	1.2	2.0	mA
	Idle		—	0.01	5.0	μA

Rev. 3.00 Mar 21, 2006 pag

REJ03

RENESAS

		2.0	—	5.5	Io
RAM standby voltage	V_{RAM}	2.0	—	—	V

Notes: 1. Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 0 V to 5.5 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or by connection to ground (GND) by the same method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. Current dissipation values are for $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CCB} - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$ with all output pins unloaded and the on-chip pull-up MOSs in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 4.5 \text{ V}$, $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CCB} - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$.
4. The port A characteristics depend on V_{CCB} , and the other pins characteristics depend on V_{CC} .

trigger input voltage	= 00) ^{*2 *6} , KIN15 to KIN8 ^{*7 *8} , IRQ2 to IRQ0 ^{*3} , IRQ5 to IRQ3	V_T^+	—	—	$V_{CC} \times 0.7$	V
		$V_T^+ - V_T^-$	0.4	—	$V_{CC} B \times 0.7$	
		V_T^-	0.8	—	—	V
		V_T^+	—	—	$V_{CC} \times 0.7$	
		$V_T^+ - V_T^-$	0.3	—	$V_{CC} B \times 0.7$	
		V_T^-	$V_{CC} \times 0.3$	—	—	V
		V_T^+	—	—	$V_{CC} \times 0.7$	
		$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	
		V_T^-	$V_{CC} \times 0.4$	—	—	
		V_T^+	—	—	$V_{CC} \times 0.8$	
Schmitt trigger input voltage (in level switching) ^{*6}	P67 to P60 (KWUL = 01)	$V_T^+ - V_T^-$	$V_{CC} \times 0.03$	—	—	
		V_T^-	$V_{CC} \times 0.45$	—	—	
		V_T^+	—	—	$V_{CC} \times 0.9$	
	P67 to P60 (KWUL = 10)	$V_T^+ - V_T^-$	$V_{CC} \times 0.03$	—	—	
		V_T^-	$V_{CC} \times 0.45$	—	—	
		V_T^+	—	—	$V_{CC} \times 0.9$	
	P67 to P60 (KWUL = 11)	$V_T^+ - V_T^-$	0.05	—	—	
		V_T^-	$V_{CC} \times 0.45$	—	—	
		V_T^+	—	—	$V_{CC} \times 0.9$	
		$V_T^+ - V_T^-$	0.05	—	—	
Input high voltage	RES, STBY, NMI, MD1, MD0 EXTAL PA7 to PA0 ^{*7} Port 7 P97, P86, P52, P42 Input pins except (1) and (2) above	(2) V_{IH}	$V_{CC} - 0.7$	—	$V_{CC} + 0.3$	V
			$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	
			$V_{CC} B \times 0.7$	—	$V_{CC} B + 0.3$	
			2.0	—	$A V_{CC} + 0.3$	
			$V_{CC} \times 0.7$	—	5.5	
			2.0	—	$V_{CC} + 0.3$	

input pins except (1)
and (3) above

Output high voltage	All output pins (except P97, P86, P52, and P42) ^{*4 *5 *8}	V_{OH}	$V_{CC} - 0.5$	—	—	V	I_{OH}	
			$V_{CC}B - 0.5$	—	—	V	I_{OH}	
			3.5	—	—	V	I_{OH}	
			3.0	—	—	V	I_{OH}	
	P97, P86, P52, and P42 ^{*4}		1.5	—	—	V	I_{OH}	
Output low voltage	All output pins (except \overline{RESO}) ^{*5}	V_{OL}	—	—	0.4	V	I_{OL}	
			Ports 1 to 3	—	—	1.0	V	I_{OL}
			\overline{RESO}	—	—	0.4	V	I_{OL}

Notes: 1. Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 2.0 V to 5.5 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

- P67 to P60 include peripheral module inputs multiplexed on those pins.
- $\overline{IRQ2}$ includes the \overline{ADTRG} signal multiplexed on that pin.
- P52/SCK0/SCL0, P97/SDA0, P86/SCK1/SCL1, P42/SCK2/SDA1, and port C high levels are NMOS push-pull outputs.

When the SCL0, SDA0, SCL1, or SDA1 (ICE = 1) pin is used as an output, it is configured as an open-drain output. Therefore, an external pull-up resistor must be connected to the output high level.

P52/SCK0, P97, P86/SCK1, P42/SCK2 (ICE = 0), and port G high levels are NMOS.

- when a pin is in output mode, the output voltage is equivalent to the applied voltage.
8. The port A characteristics depend on V_{CC} , and the other pins characteristics depend on V_{CC} .

leakage current	STBY, NMI, MD1, MD0		—	—	1.0	
	Port 7		—	—	1.0	
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A ^{*4} , B	$ I_{TSI} $	—	—	1.0	μA
Input pull-up MOS current	Ports 1 to 3	$-I_p$	30	—	300	μA
	Ports A ^{*4} , B, 6 (P6PUE = 0)		60	—	600	
	Port 6 (P6PUE = 1)		15	—	200	
	Ports 1 to 3		20	—	200	μA
	Ports A ^{*4} , B, 6 (P6PUE = 0)		40	—	500	
	Port 6 (P6PUE = 1)		10	—	150	
Input capacitance	RES (4)	C_{in}	—	—	80	pF
	NMI		—	—	50	
	P52, P97, P42, P86, PA7 to PA2		—	—	20	
	Input pins except (4) above		—	—	15	
Current dissipation ^{*2}	Normal operation	I_{CC}	—	45	58	mA
	Sleep mode		—	30	46	mA
	Standby mode ^{*3}		—	1.0	5.0	μA
			—	—	20.0	

supply current	conversion		—	0.01	5.0	μA
	Idle					
Analog power supply voltage ^{*1}	AV _{CC}	4.0	—	5.5	V	
		2.0	—	5.5		
RAM standby voltage	V _{RAM}	2.0	—	—	V	

Notes: 1. Do not leave the AV_{CC}, AV_{ref}, and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 2.0 V to 5.5 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}) or other method. Ensure that AV_{ref} ≤ AV_{CC}.

- Current dissipation values are for V_{IH} min = V_{CC} - 0.2 V, V_{CCB} - 0.2 V, and V_{CC} - 0.2 V with all output pins unloaded and the on-chip pull-up MOSs in the off state.
- Current dissipation values are for V_{IH} min = V_{CC} - 0.2 V, V_{CCB} - 0.2 V, and V_{CC} - 0.2 V with all output pins unloaded and the on-chip pull-up MOSs in the off state.
- Current dissipation values are for V_{IH} min = V_{CC} - 0.2 V, V_{CCB} - 0.2 V, and V_{CC} - 0.2 V with all output pins unloaded and the on-chip pull-up MOSs in the off state.
- The port A characteristics depend on VCCB, and the other pins characteristics depend on VCC.

voltage	KIN15 to KIN8 ^{*1,2} , IRQ2 to IRQ0 ^{*3} , IRQ5 to IRQ3	V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$		
		$V_T^+ - V_T^-$	$V_{CC} \times 0.05$ $V_{CC}B \times 0.05$	—	—		
Schmitt trigger input voltage (in level switching) ^{*6}	P67 to P60 (KWUL = 01)	V_T^-	$V_{CC} \times 0.3$	—	—	V	
		V_T^+	—	—	$V_{CC} \times 0.7$		
	P67 to P60 (KWUL = 10)	$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—		
		V_T^-	$V_{CC} \times 0.4$	—	—		
	P67 to P60 (KWUL = 11)	V_T^+	—	—	$V_{CC} \times 0.8$		
		$V_T^+ - V_T^-$	$V_{CC} \times 0.03$	—	—		
		V_T^-	$V_{CC} \times 0.45$	—	—		
		V_T^+	—	—	$V_{CC} \times 0.9$		
			$V_T^+ - V_T^-$	0.05	—	—	
	Input high voltage	RES, STBY, NMI, MD1, MD0	(2) V_{IH}	$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	V
EXTAL			$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$		
PA7 to PA0 ^{*7}			$V_{CC}B \times 0.7$	—	$V_{CC}B + 0.3$		
Port 7			$V_{CC} \times 0.7$	—	$AV_{CC} + 0.3$		
P97, P86, P52, P42			$V_{CC} \times 0.7$	—	5.5		
Input pins except (1) and (2) above			$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$		

	NMI, EXTAL, input pins except (1) and (3) above		-0.3	—	$V_{CC} \times 0.2$	V	
Output high voltage	All output pins (except P97, P86, P52, and P42) ^{*4 *5 *8}	V_{OH}	$V_{CC} - 0.5$	—	—	V	
			$V_{CC}B - 0.5$				
			$V_{CC} - 1.0$	—	—	V	
			$V_{CC}B - 1.0$				
	P97, P86, P52, and P42 ^{*4}		0.5	—	—	V	
Output low voltage	All output pins (except \overline{RESO}) ^{*5}	V_{OL}	—	—	0.4	V	
			Ports 1 to 3	—	—	1.0	V
			\overline{RESO}	—	—	0.4	V

Notes: 1. Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 0 V to 3.6 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or by connection to ground (AV_{SS}) method. Ensure that $AV_{ref} \leq AV_{CC}$.

- P67 to P60 include peripheral module inputs multiplexed on those pins.
- $\overline{IRQ2}$ includes the \overline{ADTRG} signal multiplexed on that pin.
- P52/SCK0/SCL0, P97/SDA0, P86/SCK1/SCL1, P42/SCK2/SDA1, and port G include peripheral module inputs multiplexed on those pins. P52/SCK0, P97, P86/SCK1, P42/SCK2 (ICE = 0), and port G high levels are NMOS push-pull outputs.

When the SCL0, SDA0, SCL1, or SDA1 (ICE = 1) pin is used as an output, it is an open-drain output. Therefore, an external pull-up resistor must be connected to the output high level.

P52/SCK0, P97, P86/SCK1, P42/SCK2 (ICE = 0), and port G high levels are NMOS.

When the SCK0, SCK1, or SCK 2 pin is used as an output, an external pull-up resistor must be connected in order to output high level.

- On V_{CC} .
9. For flash memory programming/erasure, the applicable range is $V_{CC} = 3.0\text{ V}$

current	MD0		—	—	1.0	
	Port 7		—	—	1.0	
Three-state leakage current (off state)	Ports 1 to 6 Ports 8, 9, A* ⁴ , B	$ I_{TSI} $	—	—	1.0	μA
Input pull-up MOS current	Ports 1 to 3	$-I_P$	5	—	150	μA
	Ports A* ⁴ , B, 6 (P6PUE = 0)		30	—	300	
	Port 6 (P6PUE = 1)		3	—	100	
Input capacitance	$\overline{\text{RES}}$ (4)	C_{in}	—	—	80	μF
	NMI		—	—	50	
	P52, P97, P42, P86, PA7 to PA2		—	—	20	
	Input pins except (4) above		—	—	15	
Current dissipation* ²	Normal operation	I_{CC}	—	30	40	mA
	Sleep mode		—	20	32	mA
	Standby mode* ³		—	1.0	5.0	μA
			—	—	20.0	
Analog power supply current	During A/D, D/A conversion	$A I_{CC}$	—	1.2	2.0	mA
	Idle		—	0.01	5.0	μA

		2.0	—	3.6	Io
RAM standby voltage	V_{RAM}	2.0	—	—	V

Notes: 1. Do not leave the AV_{CC} , AV_{ref} , and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in the range of 2.0 V to 3.6 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}), or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

- Current dissipation values are for $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CCB} - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$ with all output pins unloaded and the on-chip pull-up MOSs in the off state.
- The values are for $V_{RAM} \leq V_{CC} < 2.7 \text{ V}$, $V_{IH} \text{ min} = V_{CC} - 0.2 \text{ V}$, $V_{CCB} - 0.2 \text{ V}$, and $V_{IL} \text{ max} = 0.2 \text{ V}$.
- The port A characteristics depend on V_{CCB} , and the other pins characteristics depend on V_{CC} .
- For flash memory programming/erasure, the applicable range is $V_{CC} = 3.0 \text{ V}$.

	PB1, PB0					
Input low voltage	P37 to P30, P83 to P80, PB1, PB0	V_{IL}	—	$V_{CC} \times 0.3$	V	
Output high voltage	P37, P33 to P30, P82 to P80, PB1, PB0	V_{OH}	$V_{CC} \times 0.9$	—	V	$I_{OH} = -$
Output low voltage	P37, P33 to P30, P82 to P80, PB1, PB0	V_{OL}	—	$V_{CC} \times 0.1$	V	$I_{OL} = 1$

Note: * Do not leave the AV_{CC} , AV_{ref} and AV_{SS} pins open even if the A/D converter and D/A converter are not used.

Even if the A/D converter and D/A converter are not used, apply a value in 2.0 V to 3.6 V to AV_{CC} and AV_{ref} pins by connection to the power supply (V_{CC}) or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

	PS2AD to PS2CD, PA7 to PA4 (bus drive function selected)				
	Ports 1 to 3		—	—	10
	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1 to 3	$\sum I_{OL}$	—	—	80
	Total of all output pins, including the above		—	—	120
Permissible output high current (per pin)	All output pins	$-I_{OH}$	—	—	2
Permissible output high current (total)	Total of all output pins	$\sum -I_{OH}$	—	—	40

	<u>RES0</u>		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1 to 3	$\sum I_{OL}$	—	—	40
	Total of all output pins, including the above		—	—	60
Permissible output high current (per pin)	All output pins	$-I_{OH}$	—	—	2
Permissible output high current (total)	Total of all output pins	$\sum -I_{OH}$	—	—	30

- Notes: 1. To protect chip reliability, do not exceed the output current values in table 2.
2. When driving a Darlington pair or LED, always insert a current-limiting resistor in series with the output line, as shown in figures 27.1 and 27.2.

	$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—		
Input high voltage	V_{IH}	$V_{CC} \times 0.7$	—	5.5	V	
Input low voltage	V_{IL}	-0.5	—	$V_{CC} \times 0.3$		
Output low voltage	V_{OL}	—	—	0.8	V	$I_{OL} = 16 \text{ mA}$ V to 5.5 V
				0.5		$I_{OL} = 8 \text{ mA}$
				0.4		$I_{OL} = 3 \text{ mA}$
Input capacitance	C_{in}	—	—	20	pF	$V_{in} = 0 \text{ V}$, f $T_a = 25^\circ\text{C}$
Three-state leakage current (off state)	$ I_{TSL} $	—	—	1.0	μA	$V_{in} = 0.5 \text{ to } 5.5 \text{ V}$
SCL, SDA output fall time	t_{of}	$20 + 0.1 C_b$	—	250	ns	

Conditions: $V_{CC} = 4.0 \text{ V to } 5.5 \text{ V}$, $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ (3-V version product),
 $V_{CCB} = 2.7 \text{ V to } 5.5 \text{ V}$, $V_{SS} = 0 \text{ V}$

Applicable Pins: PS2AC, PS2AD, PS2BC, PS2BD, PS2CC, PS2CD, PA7 to PA4 (but function selected)

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Output low voltage	V_{OL}	—	—	0.8	V	$I_{OL} = 16 \text{ mA}$, f $V_{CCB} = 4.5 \text{ V to } 5.5 \text{ V}$
				0.5		$I_{OL} = 8 \text{ mA}$
				0.4		$I_{OL} = 3 \text{ mA}$

Table 27.20 Clock Timing

Condition A: $V_{CC} = 5.0 \text{ V} \pm 10\%$, $V_{CCB} = 5.0 \text{ V} \pm 10\%$, $V_{SS} = 0 \text{ V}$, $\phi = 2 \text{ MHz}$ to maximum operating frequency, $T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition B: $V_{CC} = 4.0 \text{ V}$ to 5.5 V , $V_{CCB} = 4.0 \text{ V}$ to 5.5 V , $V_{SS} = 0 \text{ V}$, $\phi = 2 \text{ MHz}$ to maximum operating frequency, $T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7 \text{ V}$ to 3.6 V , $V_{CCB} = 2.7 \text{ V}$ to 5.5 V , $V_{SS} = 0 \text{ V}$, $\phi = 2 \text{ MHz}$ to maximum operating frequency, $T_a = -20$ to $+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit
		10 MHz		16 MHz		20 MHz		
		Min	Max	Min	Max	Min	Max	
Clock cycle time	t_{cyc}	100	500	62.5	500	50	500	ns
Clock high pulse width	t_{CH}	30	—	20	—	17	—	ns
Clock low pulse width	t_{CL}	30	—	20	—	17	—	ns
Clock rise time	t_{Cr}	—	20	—	10	—	8	ns
Clock fall time	t_{Cf}	—	20	—	10	—	8	ns
Oscillation settling time at reset (crystal)	t_{OSC1}	20	—	10	—	10	—	ms
Oscillation settling time in software standby (crystal)	t_{OSC2}	8	—	8	—	8	—	ms
External clock output stabilization delay time	t_{DEXT}	500	—	500	—	500	—	μs

Condition B: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{CCB} = 4.0\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$,
 2 MHz to maximum operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$ (normal specification product), $T_a = -40\text{ to }+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{CCB} = 2.7\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$,
 2 MHz to maximum operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit
		10 MHz		16 MHz		20 MHz		
		Min	Max	Min	Max	Min	Max	
$\overline{\text{RES}}$ setup time	t_{RESS}	300	—	200	—	200	—	ns
$\overline{\text{RES}}$ pulse width	t_{RESW}	20	—	20	—	20	—	t_{cyc}
NMI setup time (NMI)	t_{NMIS}	250	—	150	—	150	—	ns
NMI hold time (NMI)	t_{NMIH}	10	—	10	—	10	—	ns
NMI pulse width (NMI) (exiting software standby mode)	t_{NMIW}	200	—	200	—	200	—	ns
IRQ setup time ($\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$)	t_{IRQS}	250	—	150	—	150	—	ns
IRQ hold time ($\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$)	t_{IRQH}	10	—	10	—	10	—	ns
IRQ pulse width ($\overline{\text{IRQ7}}$, $\overline{\text{IRQ6}}$, $\overline{\text{IRQ2}}$ to $\overline{\text{IRQ0}}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	200	—	ns

Condition B: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{CCB} = 4.0\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to }10\text{ MHz}$
operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$ (normal specification product),
 $T_a = -40\text{ to }+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{CCB} = 2.7\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to }10\text{ MHz}$
operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit
		10 MHz		16 MHz		20 MHz		
		Min	Max	Min	Max	Min	Max	
Address delay time	t_{AD}	—	40	—	30	—	20	ns
Address setup time	t_{AS}	$0.5 \times t_{cyc} - 30$	—	$0.5 \times t_{cyc} - 20$	—	$0.5 \times t_{cyc} - 15$	—	ns
Address hold time	t_{AH}	$0.5 \times t_{cyc} - 20$	—	$0.5 \times t_{cyc} - 15$	—	$0.5 \times t_{cyc} - 10$	—	ns
\overline{CS} delay time (\overline{IOS})	t_{CSD}	—	40	—	30	—	20	ns
\overline{AS} delay time	t_{ASD}	—	60	—	45	—	30	ns
\overline{RD} delay time 1	t_{RSD1}	—	60	—	45	—	30	ns
\overline{RD} delay time 2	t_{RSD2}	—	60	—	45	—	30	ns
Read data setup time	t_{RDS}	35	—	20	—	15	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns
Read data access time 1	t_{ACC1}	—	$1.0 \times t_{cyc} - 60$	—	$1.0 \times t_{cyc} - 40$	—	$1.0 \times t_{cyc} - 30$	ns
Read data access time 2	t_{ACC2}	—	$1.5 \times t_{cyc} - 50$	—	$1.5 \times t_{cyc} - 35$	—	$1.5 \times t_{cyc} - 25$	ns

Rev. 3.00 Mar 21, 2006 pag

REJ03

RENESAS

access time 5	ACCS		$t_{cyc} - 60$		$t_{cyc} - 40$		$t_{cyc} - 30$	
HWR, LWR delay time 1	t_{WRD1}	—	60	—	45	—	30	ns
HWR, LWR delay time 2	t_{WRD2}	—	60	—	45	—	30	ns
HWR, LWR pulse width 1	t_{WSW1}	$1.0 \times$ $t_{cyc} - 40$	—	$1.0 \times$ $t_{cyc} - 30$	—	$1.0 \times$ $t_{cyc} - 20$	—	ns
HWR, LWR pulse width 2	t_{WSW2}	$1.5 \times$ $t_{cyc} - 40$	—	$1.5 \times$ $t_{cyc} - 30$	—	$1.5 \times$ $t_{cyc} - 20$	—	ns
Write data delay time	t_{WDD}	—	60	—	45	—	30	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	20	—	15	—	10	—	ns
WAIT setup time	t_{WTS}	60	—	45	—	30	—	ns
WAIT hold time	t_{WTH}	10	—	5	—	5	—	ns

Condition C: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{CCB} = 2.7\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to }n$
operating frequency, $T_a = -20\text{ to }+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit
		10 MHz		16 MHz		20 MHz		
		Min	Max	Min	Max	Min	Max	
Address delay time	t_{AD}	—	60	—	45	—	30	ns
Address setup time	t_{AS}	$0.5 \times t_{cyc} - 50$	—	$0.5 \times t_{cyc} - 35$	—	$0.5 \times t_{cyc} - 25$	—	ns
Address hold time	t_{AH}	$0.5 \times t_{cyc} - 20$	—	$0.5 \times t_{cyc} - 15$	—	$0.5 \times t_{cyc} - 10$	—	ns
\overline{CS} delay time (\overline{IOS})	t_{CSD}	—	60	—	45	—	30	ns
\overline{AS} delay time	t_{ASD}	—	60	—	45	—	30	ns
\overline{RD} delay time 1	t_{RSD1}	—	60	—	45	—	30	ns
\overline{RD} delay time 2	t_{RSD2}	—	60	—	45	—	30	ns
Read data setup time	t_{RDS}	35	—	20	—	15	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns
Read data access time 1	t_{ACC1}	—	$1.0 \times t_{cyc} - 80$	—	$1.0 \times t_{cyc} - 55$	—	$1.0 \times t_{cyc} - 40$	ns
Read data access time 2	t_{ACC2}	—	$1.5 \times t_{cyc} - 50$	—	$2.5 \times t_{cyc} - 35$	—	$2.5 \times t_{cyc} - 25$	ns
Read data access time 3	t_{ACC3}	—	$2.0 \times t_{cyc} - 80$	—	$3.0 \times t_{cyc} - 55$	—	$3.0 \times t_{cyc} - 40$	ns

Rev. 3.00 Mar 21, 2006 pag

REJ03

RENESAS

delay time 1								
HWR, LWR delay time 2	t_{WRD2}	—	60	—	45	—	30	ns
HWR, LWR pulse width 1	t_{WSW1}	$1.0 \times$ $t_{cyc} - 40$	—	$1.0 \times$ $t_{cyc} - 30$	—	$1.0 \times$ $t_{cyc} - 20$	—	ns
HWR, LWR pulse width 2	t_{WSW2}	$1.5 \times$ $t_{cyc} - 40$	—	$1.5 \times$ $t_{cyc} - 30$	—	$1.5 \times$ $t_{cyc} - 20$	—	ns
Write data delay time	t_{WDD}	—	60	—	45	—	30	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	20	—	15	—	10	—	ns
WAIT setup time	t_{WTS}	60	—	45	—	30	—	ns
WAIT hold time	t_{WTH}	10	—	5	—	5	—	ns

$T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition B: $V_{CC} = 4.0$ V to 5.5 V, $V_{CCB} = 4.0$ V to 5.5 V, $V_{SS} = 0$ V,
 $\phi = 32.768$ kHz*, 2 MHz to maximum operating frequency,
 $T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7$ V to 3.6 V, $V_{CCB} = 2.7$ V to 5.5 V, $V_{SS} = 0$ V, $\phi = 32.768$ kHz
 2 MHz to maximum operating frequency, $T_a = -20$ to $+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit	
		10 MHz		16 MHz		20 MHz			
		Min	Max	Min	Max	Min	Max		
I/O ports	Output data delay time	t_{PVD}	—	100	—	50	—	50	ns
	Input data setup time	t_{PRS}	50	—	30	—	30	—	
	Input data hold time	t_{PRH}	50	—	30	—	30	—	
FRT	Timer output delay time	t_{FTOD}	—	100	—	50	—	50	ns
	Timer input setup time	t_{FTIS}	50	—	30	—	30	—	
	Timer clock input setup time	t_{FTCS}	50	—	30	—	30	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	1.5	—
Both edges		t_{FTCWL}	2.5	—	2.5	—	2.5	—	
TMR	Timer output delay time	t_{TMOD}	—	100	—	50	—	50	ns
	Timer reset input setup time	t_{TMRS}	50	—	30	—	30	—	
	Timer clock input setup time	t_{TMCS}	50	—	30	—	30	—	

PWMX

SCI	Input clock cycle	Asynchronous	t_{Soyc}	4	—	4	—	4	—	t_{Soyc}
		Synchronous		6	—	6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6	t_{Soyc}
	Input clock rise time		t_{SCKr}	—	1.5	—	1.5	—	1.5	t_{cyc}
	Input clock fall time		t_{SCKf}	—	1.5	—	1.5	—	1.5	
	Transmit data delay time (clocked synchronous)		t_{TXD}	—	100	—	50	—	50	ns
	Receive data setup time (clocked synchronous)		t_{RXS}	100	—	50	—	50	—	
	Receive data hold time (clocked synchronous)		t_{RXH}	100	—	50	—	50	—	
A/D converter	Trigger input setup time		t_{TRGS}	50	—	30	—	30	—	ns
WDT	\overline{RESO} output delay time		t_{RESD}	—	200	—	120	—	100	ns
	\overline{RESO} output pulse width		t_{RESOW}	132	—	132	—	132	—	t_{cyc}

Note: * Only peripheral modules that can be used in subclock operation

$T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{cc} = 2.7$ V to 3.6 V, $V_{ccB} = 2.7$ V to 5.5 V, $V_{ss} = 0$ V, $\phi = 2$ MHz to ∞
operating frequency, $T_a = -20$ to $+75^\circ\text{C}$

Item	Symbol	Condition A		Condition B		Condition C		Unit	
		10 MHz		16 MHz		20 MHz			
		Min	Max	Min	Max	Min	Max		
HIF read cycle	$\overline{\text{CS}}/\text{HA0}$ setup time	t_{HAR}	10	—	10	—	10	—	ns
	$\overline{\text{CS}}/\text{HA0}$ hold time	t_{HRA}	10	—	10	—	10	—	
	$\overline{\text{IOR}}$ pulse width	t_{HRPW}	220	—	120	—	120	—	
	HDB delay time	t_{HRD}	—	200	—	100	—	100	
	HDB hold time	t_{HRF}	0	40	0	25	0	25	
	HIRQ delay time	t_{HIRQ}	—	200	—	120	—	120	
HIF write cycle	$\overline{\text{CS}}/\text{HA0}$ setup time	t_{HAW}	10	—	10	—	10	—	
	$\overline{\text{CS}}/\text{HA0}$ hold time	t_{HWA}	10	—	10	—	10	—	
	$\overline{\text{IOW}}$ pulse width	t_{HWPW}	100	—	60	—	60	—	
	HDB setup time	Fast A20 gate not used	t_{HDW}	50	—	30	—	30	—
		Fast A20 gate used		85	—	55	—	45	—
	HDB hold time	t_{HWD}	25	—	15	—	15	—	
	GA20 delay time	t_{HGA}	—	180	—	90	—	90	

KCLK, KD input data hold time	t_{KBIS}	150	—	—	ns
KCLK, KD output delay time	t_{KBOD}	—	—	450	ns
KCLK, KD capacitive load	C_b	—	—	400	pF

Table 27.25 I²C Bus Timing

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ (3-V product), $V_{SS} = 0\text{ V}$,
 $\phi = 5\text{ MHz to maximum operating frequency}$,

Item	Symbol	Ratings			Unit	Test Conditions
		Min	Typ	Max		
SCL input cycle time	t_{SCL}	12	—	—	t_{cyc}	
SCL input high pulse width	t_{SCLH}	3	—	—	t_{cyc}	
SCL input low pulse width	t_{SCLL}	5	—	—	t_{cyc}	
SCL, SDA input rise time	t_{Sr}	—	—	7.5*	t_{cyc}	
SCL, SDA input fall time	t_{Sf}	—	—	300	ns	
SCL, SDA input spike pulse elimination time	t_{SP}	—	—	1	t_{cyc}	
SDA input bus free time	t_{BUF}	5	—	—	t_{cyc}	
Start condition input hold time	t_{STAH}	3	—	—	t_{cyc}	
Retransmission start condition input setup time	t_{STAS}	3	—	—	t_{cyc}	
Stop condition input setup time	t_{STOS}	3	—	—	t_{cyc}	
Data input setup time	t_{SDAS}	0.5	—	—	t_{cyc}	
Data input hold time	t_{SDAH}	0	—	—	ns	
SCL, SDA capacitive load	C_b	—	—	400	pF	

Note: * $17.5t_{cyc}$ can be set according to the clock selected for use by the I²C module. see section 16.6, Usage Notes.

Input clock pulse width (L)	t_{LCKL}	11	—	—
Transmit signal delay time	t_{TXD}	2	—	11
Transmit signal floating delay time	t_{OFF}	—	—	28
Receive signal setup time	t_{RXS}	7	—	—
Receive signal hold time	t_{RXH}	0	—	—

$T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition B: $V_{CC} = 4.0$ V to 5.5 V, $AV_{CC} = 4.0$ V to 5.5 V, $AV_{ref} = 4.0$ V to AV_{CC} ,
 $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ MHz to maximum operating frequency,
 $T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7$ V to 3.6 V, $AV_{CC} = 2.7$ V to 3.6 V, $AV_{ref} = 2.7$ V to AV_{CC} ,
 $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ MHz to maximum operating frequency,
 $T_a = -20$ to $+75^\circ\text{C}$

Item	Condition C			Condition B			Condition A		
	10 MHz			16 MHz			20 MHz		
	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max
Resolution	10	10	10	10	10	10	10	10	10
Conversion time ^{*3}	—	—	13.4	—	—	8.4	—	—	6.7
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	5	—	—	10^{*1} 5^{*2}	—	—	10^{*1} 5^{*2}
Nonlinearity error	—	—	± 7.0	—	—	± 3.0	—	—	± 3.0
Offset error	—	—	± 7.5	—	—	± 3.5	—	—	± 3.5
Full-scale error	—	—	± 7.5	—	—	± 3.5	—	—	± 3.5
Quantization error	—	—	± 0.5	—	—	± 0.5	—	—	± 0.5
Absolute accuracy	—	—	± 8.0	—	—	± 4.0	—	—	± 4.0

- Notes: 1. When conversion time $\geq 11.17 \mu\text{s}$ (CKS = 0, or $\phi \leq 12$ MHz at CKS = 1)
2. When conversion time $< 11.17 \mu\text{s}$ ($\phi > 12$ MHz at CKS = 1)
3. At the maximum operating frequency in single mode.

$V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz}$ to maximum operating frequency,
 $T_a = -20$ to $+75^\circ\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^\circ\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 3.0\text{ V}$ to 3.6 V^{*4} , $AV_{CC} = 3.0\text{ V}$ to 3.6 V^{*4} , $AV_{ref} = 3.0\text{ V}$ to AV_{CC} ,
 $V_{CCB} = 3.0\text{ V}$ to 5.5 V^{*4} , $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz}$ to maximum operating
 frequency, $T_a = -20$ to $+75^\circ\text{C}$

Item	Condition C			Condition B			Condition A		
	10 MHz			16 MHz			20 MHz		
	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max
Resolution	10	10	10	10	10	10	10	10	10
Conversion time ^{*3}	—	—	13.4	—	—	8.4	—	—	6.7
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	5	—	—	$\frac{10^{*1}}{5^{*2}}$	—	—	$\frac{10^{*1}}{5^{*2}}$
Nonlinearity error	—	—	± 11.0	—	—	± 5.0	—	—	± 5.0
Offset error	—	—	± 11.5	—	—	± 5.5	—	—	± 5.5
Full-scale error	—	—	± 11.5	—	—	± 5.5	—	—	± 5.5
Quantization error	—	—	± 0.5	—	—	± 0.5	—	—	± 0.5
Absolute accuracy	—	—	± 12.0	—	—	± 6.0	—	—	± 6.0

Notes: 1. When conversion time $\geq 11.17\ \mu\text{s}$ (CKS = 0, or $\phi \leq 12\text{ MHz}$ at CKS = 1)
 2. When conversion time $< 11.17\ \mu\text{s}$ ($\phi > 12\text{ MHz}$ at CKS = 1)
 3. At the maximum operating frequency in single mode.
 4. When using CIN, ensure that $V_{CC} = 3.0\text{ V}$ to 3.6 V , $AV_{CC} = 3.0\text{ V}$ to 3.6 V , $AV_{ref} = 3.0\text{ V}$, $V_{CCB} = 3.0\text{ V}$ to 5.5 V .

$T_a = -40$ to $+85^{\circ}\text{C}$ (wide range temperature specification product)

Condition B: $V_{CC} = 4.0$ V to 5.5 V, $AV_{CC} = 4.0$ V to 5.5 V, $AV_{ref} = 4.0$ V to AV_{CC} ,
 $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ MHz to maximum operating frequency,
 $T_a = -20$ to $+75^{\circ}\text{C}$ (normal specification product),
 $T_a = -40$ to $+85^{\circ}\text{C}$ (wide range temperature specification product)

Condition C: $V_{CC} = 2.7$ V to 3.6 V, $AV_{CC} = 2.7$ V to 3.6 V, $AV_{ref} = 2.7$ V to AV_{CC} ,
 $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ MHz to maximum operating frequency,
 $T_a = -20$ to $+75^{\circ}\text{C}$

Item		Condition C			Condition B			Condition		
		10 MHz			16 MHz			20 MHz		
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max
Resolution		8	8	8	8	8	8	8	8	8
Conversion time	With 20 pF load capacitance	—	—	10	—	—	10	—	—	10
Absolute accuracy	With 2 M Ω load resistance	—	± 2.0	± 3.0	—	± 1.0	± 1.5	—	± 1.0	± 1.5
	With 4 M Ω load resistance	—	—	± 2.0	—	—	± 1.0	—	—	± 1.0

3-V version conditions: $V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $V_{SS} = 0 \text{ V}$, $T_a = -20 \text{ to } +75^\circ\text{C}$

Item	Symbol	Min	Typ	Max	Unit	
Programming time ^{*1 *2 *4}	t_P	—	10	200	ms/ 128 bytes	
Erase time ^{*1 *3 *6}	t_E	—	100	1200	ms/block	
Reprogramming count	N_{WEC}	—	—	100	times	
Programming	Wait time after SWE-bit setting ^{*1}	x	1	—	—	μs
	Wait time after PSU-bit setting ^{*1}	y	50	—	—	μs
	Wait time after P-bit setting ^{*1 *4}	z1	28	30	32	μs
		z2	198	200	202	μs
		z3	8	10	12	μs
	Wait time after P-bit clear ^{*1}	α	5	—	—	μs
	Wait time after PSU-bit clear ^{*1}	β	5	—	—	μs
	Wait time after PV-bit setting ^{*1}	γ	4	—	—	μs
	Wait time after dummy write ^{*1}	ε	2	—	—	μs
	Wait time after PV-bit clear ^{*1}	η	2	—	—	μs
	Wait time after SWE-bit clear ^{*1}	θ	100	—	—	μs
	Maximum programming count ^{*1 *4 *5}	N	—	—	1000	times
	Erase	Wait time after SWE-bit setting ^{*1}	x	1	—	—
Wait time after ESU-bit setting ^{*1}		y	100	—	—	μs
Wait time after E-bit setting ^{*1 *6}		z	10	—	100	ms
Wait time after E-bit clear ^{*1}		α	10	—	—	μs
Wait time after ESU-bit clear ^{*1}		β	10	—	—	μs
Wait time after EV-bit setting ^{*1}		γ	20	—	—	μs
Wait time after dummy write ^{*1}		ε	2	—	—	μs
Wait time after EV-bit clear ^{*1}		η	4	—	—	μs
Wait time after SWE-bit clear ^{*1}		θ	100	—	—	μs
Maximum erase count ^{*1 *6 *7}		N	—	—	120	times

Rev. 3.00 Mar 21, 2006 pag

REJ05

RENESAS

- z1, z2 and z3 to allow programming within the maximum programming time.
- The wait time after P-bit setting (z1, z2, and z3) should be alternated according to the number of writes (n) as follows:
- $1 \leq n \leq 6$ z1 = 30 μ s, z3 = 10 μ s
 $7 \leq n \leq 1000$ z2 = 200 μ s
6. Maximum erase time (t_E (max))
 t_E (max) = Wait time after E-bit setting (z) \times maximum erase count (N)
 7. The maximum number of erases (N) should be set according to the actual system power supply to allow erasing within the maximum erase time (t_E (max)).

27.2.7 Usage Notes

1. Internal step-down products

The H8S/2148 B-masked product (HD64F2148B) includes an internal step-down circuit that can step down the microprocessor internal power supply voltage to the appropriate level. One or two (in parallel) internal voltage regulating capacitors (0.47 μ F) should be installed between the internal step-down pin (VCL pin) and VSS pin. The method of connecting an external capacitor(s) is shown in figure 27.5.

For the 5-V and 4-V version products whose power supply (VCC) voltage exceeds 5 V, do not connect the VCC power supply to the VCL pin of the internal step-down product. Connect the VCC power supply to the VCC1 pin, as usual.)

For the 3-V version product whose power supply (VCC) voltage is 3.6 V or less, connect the system power supply to the VCL pin together with the VCC1 pins.

When switching from the F-ZTAT version product without the internal step-down function to the F-ZTAT B-masked product with the internal step-down function, note that the VCC2 pin is allocated to the same location as the VCC2 pin of the product without the internal step-down function. Therefore, the difference in the circuits between before and after the switch should be considered when designing the PC board patterns.

///
The VCC power supply should not be connected to the VCL pin of the product with the internal step-down function. (Connect the VCC power supply to other VCC1 pins as usual.)

Be sure to connect power supply regulating capacitor(s) to the VCL pin. One or two (parallel) 0.47- μ F multilayer capacitors should be used near the VCL pin. For 3-V products used with the voltage of 3.6 V or less, connect the Vcc power supply in the same way as the products without the internal step-down function.

< Product with internal step-down function >

HD64F2145B

HD64F2148B

///
The VCC2 pin of the product without the internal step-down function is allocated at the same location as the product with the internal step-down function. It is recommended that a bypass capacitor be connected to the power supply pins. (The values are referred to the product without internal step-down function.)

< Product without internal step-down function >

HD64F2145BV

HD64F2148BV

Figure 27.5 Connection of VCL Capacitor

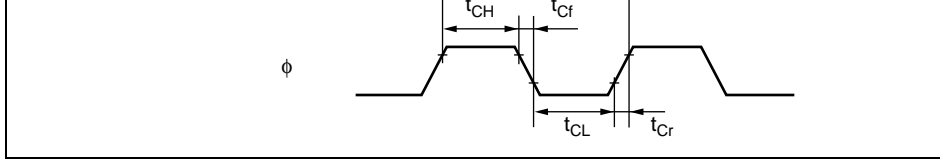


Figure 27.6 System Clock Timing

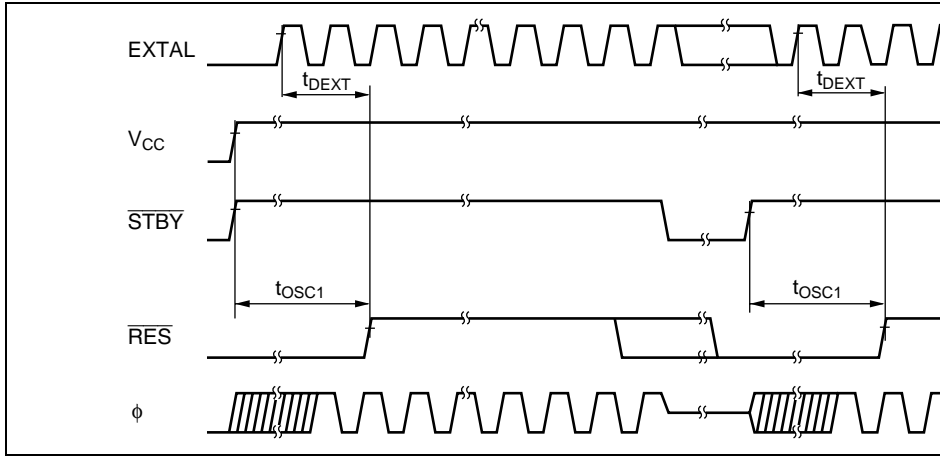


Figure 27.7 Oscillation Settling Timing

Note: $i = 0$ to 2, 6, 7

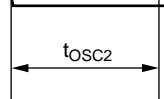


Figure 27.8 Oscillation Setting Timing (Exiting Software Standby Mode)

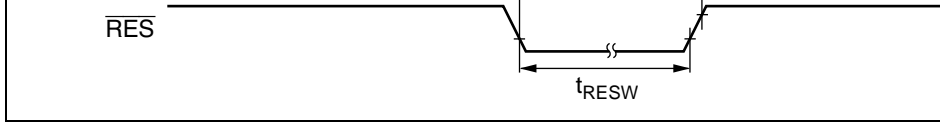


Figure 27.9 Reset Input Timing

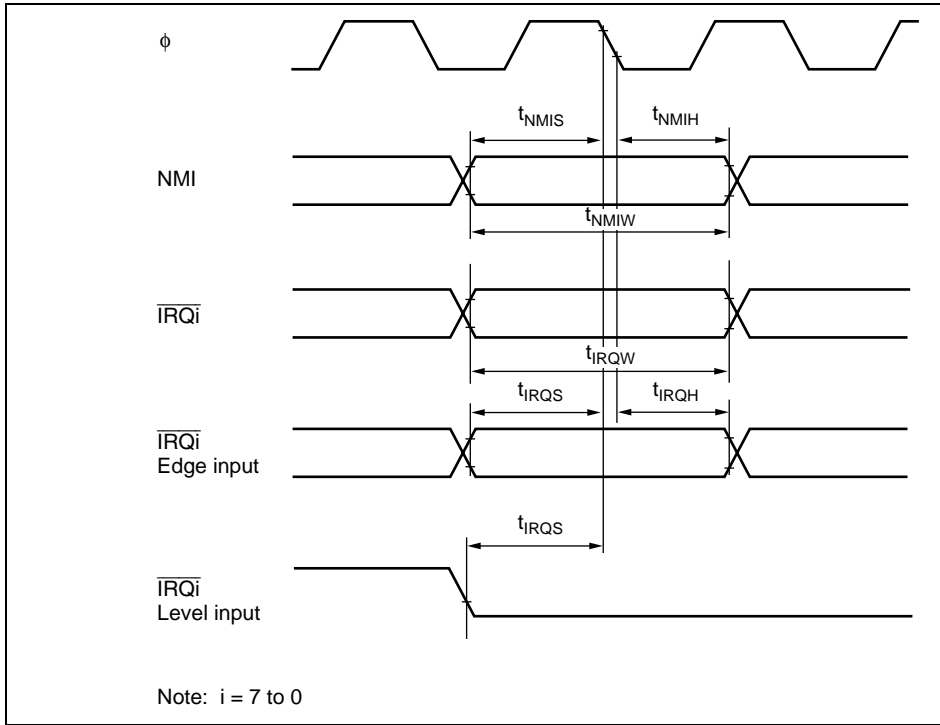
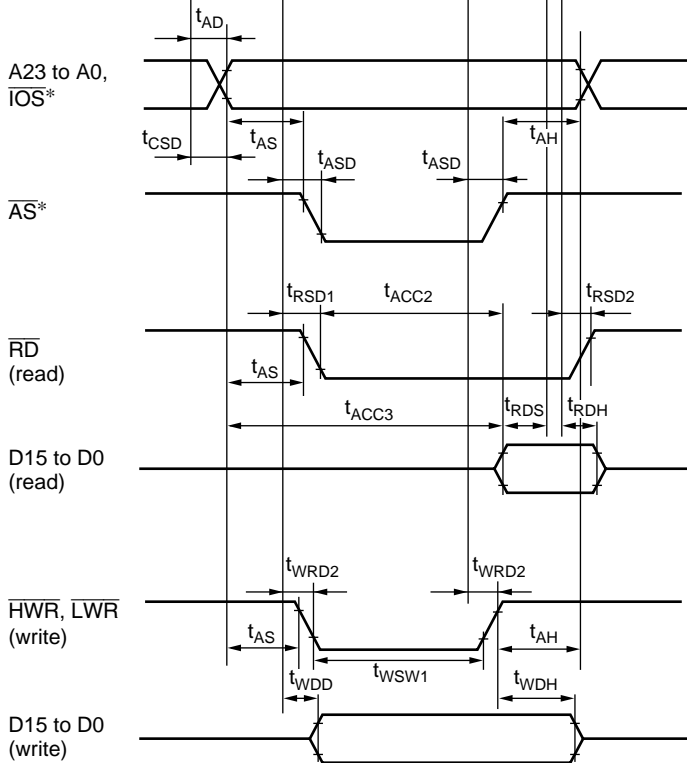
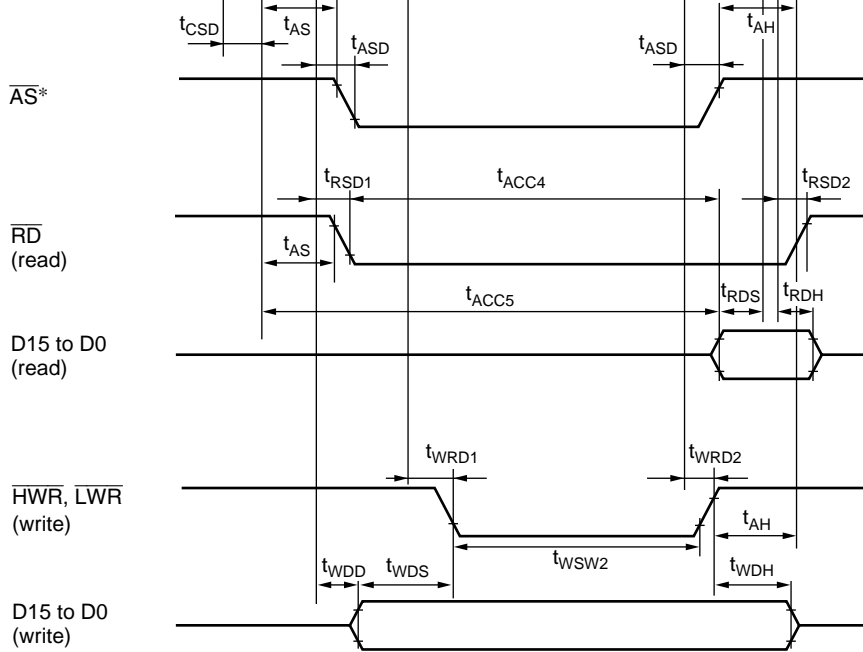


Figure 27.10 Interrupt Input Timing



Note: * \overline{AS} and \overline{IOS} are the same pin. The function is selected by the IOSE bit in SYS

Figure 27.11 Basic Bus Timing (Two-State Access)



Note: * \overline{AS} and \overline{IOS} are the same pin. The function is selected by the IOSE bit in SYSCR.

Figure 27.12 Basic Bus Timing (Three-State Access)

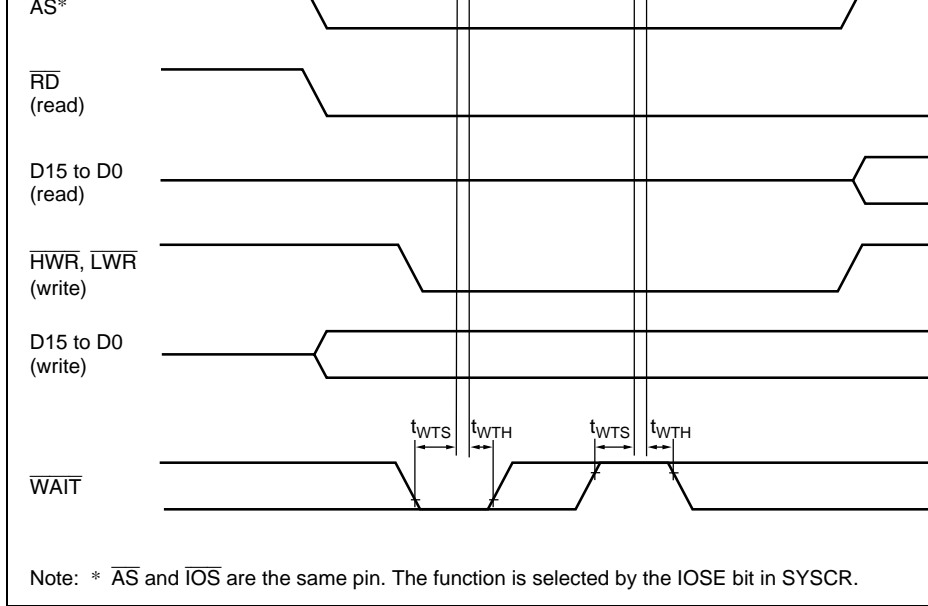


Figure 27.13 Basic Bus Timing (Three-State Access with One Wait State)

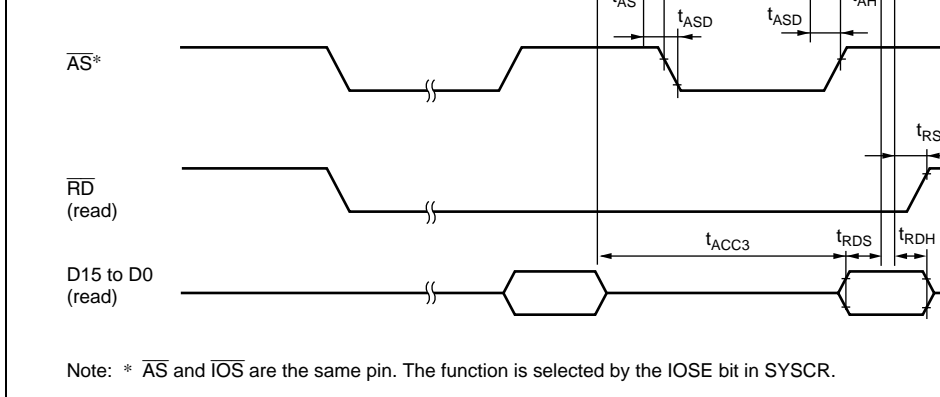
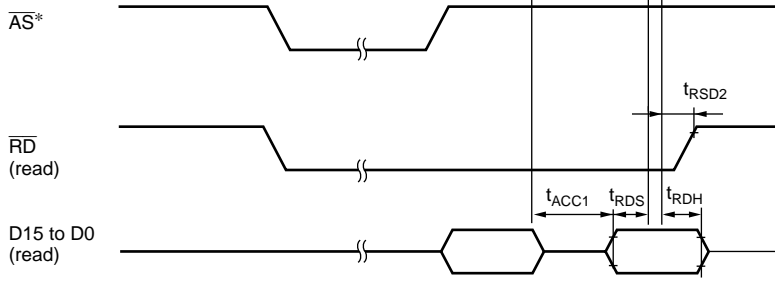


Figure 27.14 Burst ROM Access Timing (Two-State Access)



Note: * \overline{AS} and \overline{IOS} are the same pin. The function is selected by the IOSE bit in SYSCR.

Figure 27.15 Burst ROM Access Timing (One-State Access)

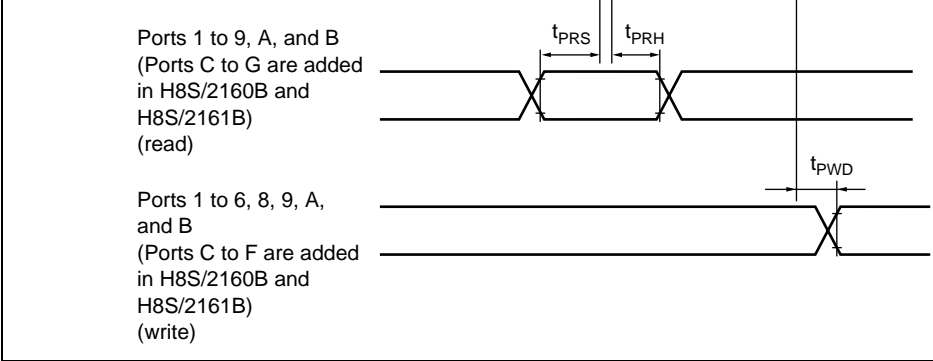


Figure 27.16 I/O Port Input/Output Timing

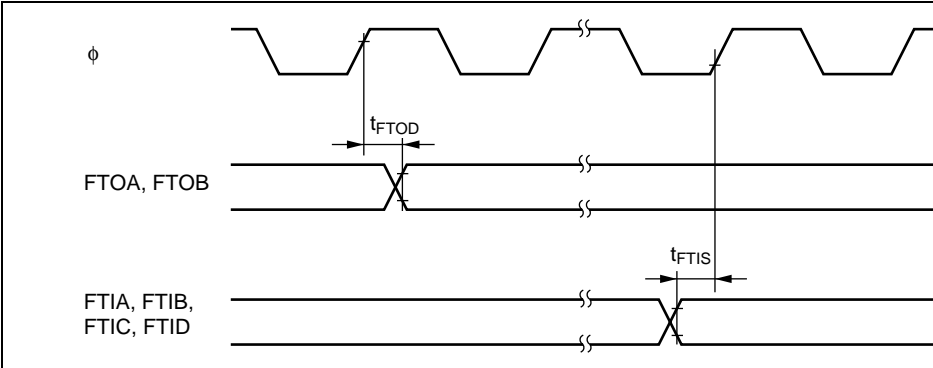


Figure 27.17 FRT Input/Output Timing

Figure 27.18 FRT Clock Input Timing

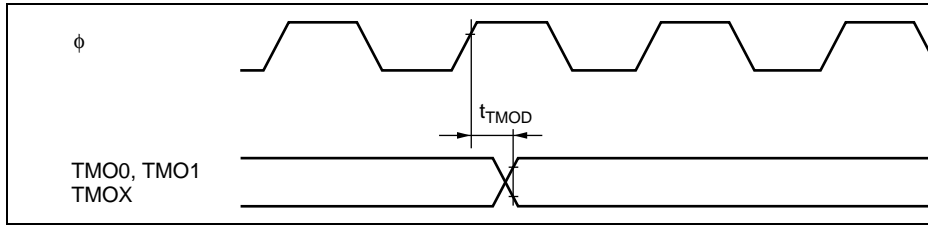


Figure 27.19 8-Bit Timer Output Timing

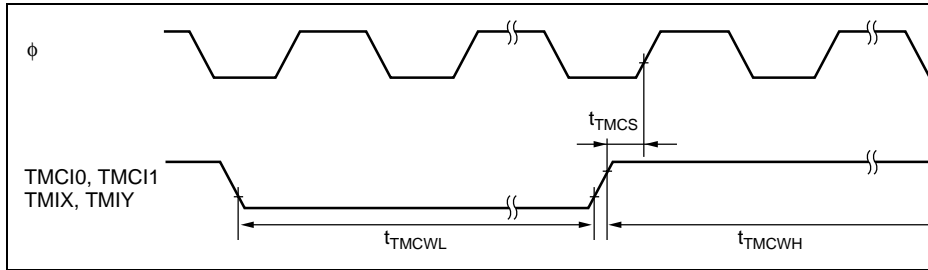


Figure 27.20 8-Bit Timer Clock Input Timing

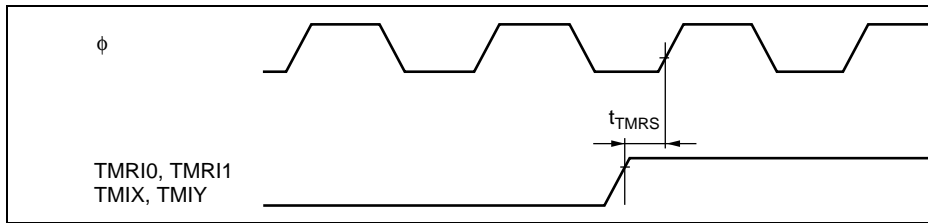


Figure 27.21 8-Bit Timer Reset Input Timing

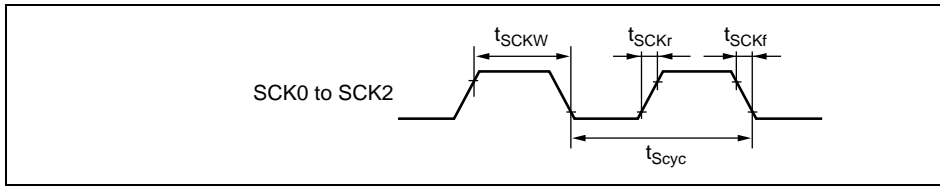


Figure 27.23 SCK Clock Input Timing

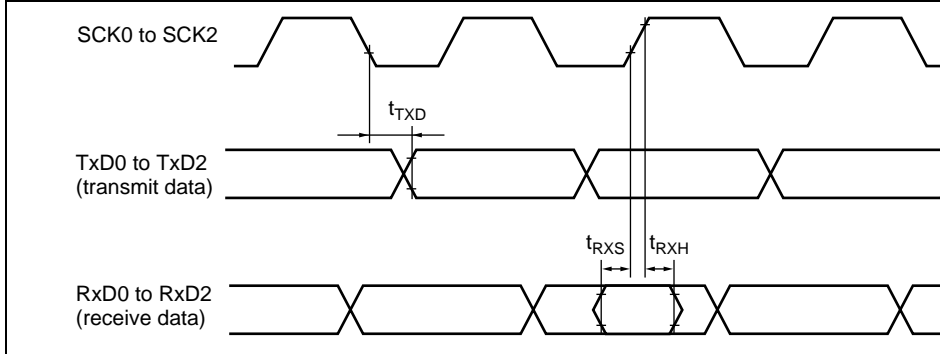


Figure 27.24 SCI Input/Output Timing (Synchronous Mode)

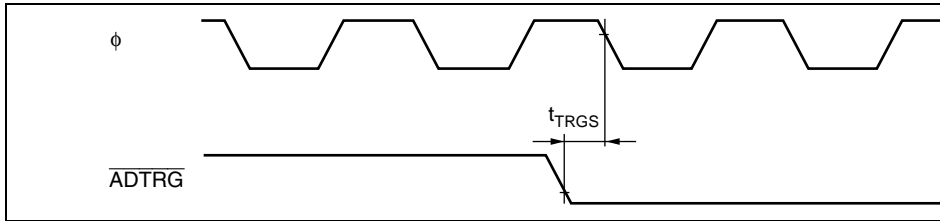


Figure 27.25 A/D Converter External Trigger Input Timing

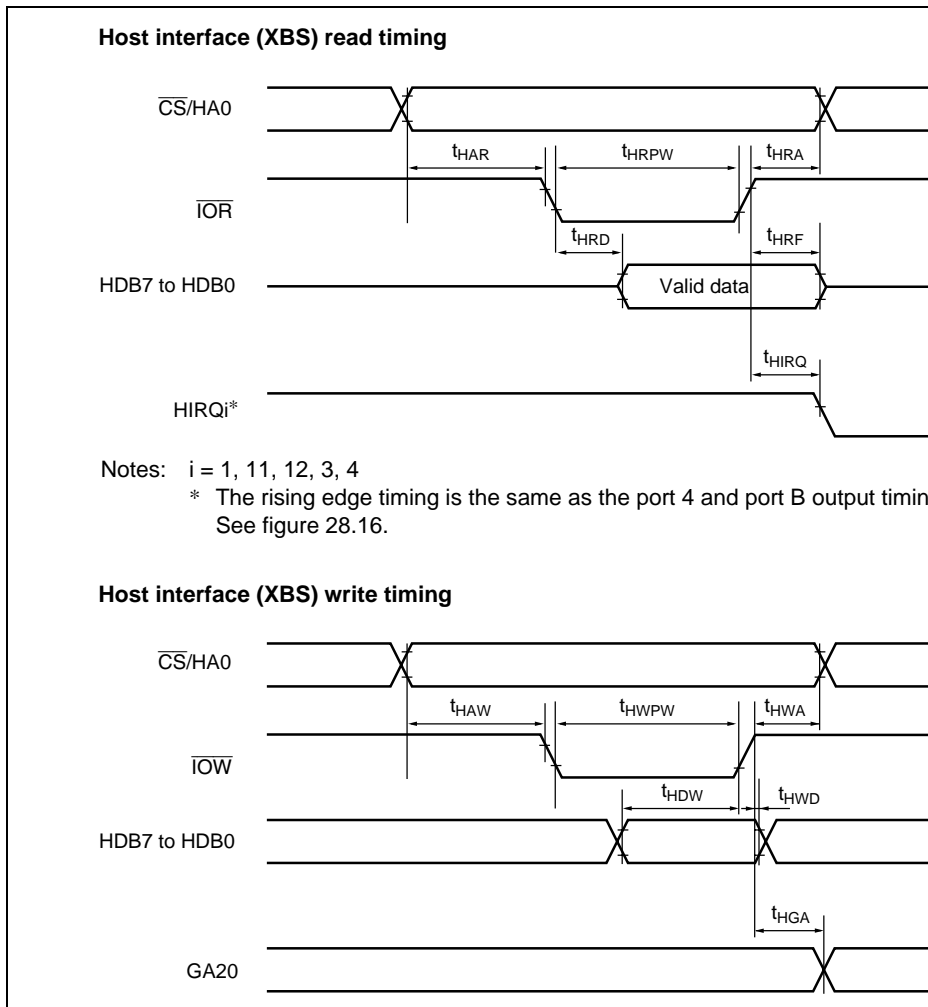
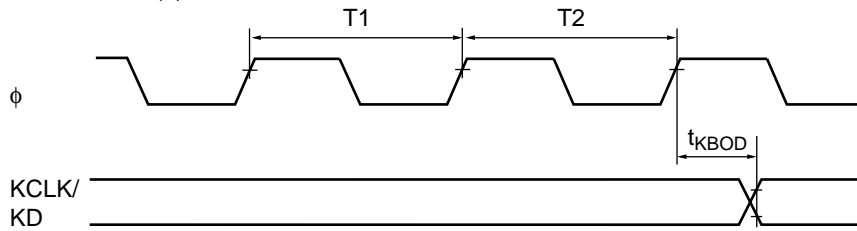
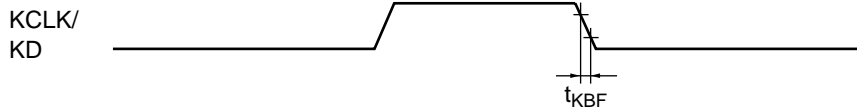


Figure 27.27 Host Interface (XBS) Timing

2. Transmission (a)



Transmission (b)



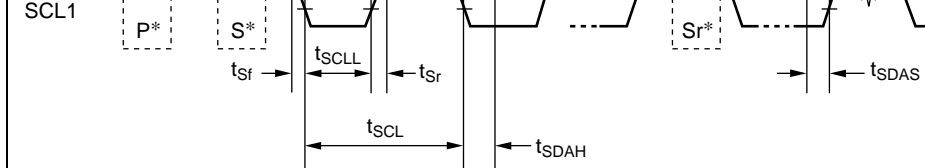
Legend:

KCLK: PS2AC to PS2CC

KD: PS2AD to PS2CD

Note: ϕ shown here is the clock scaled by $1/N$ when the operating mode is active medium-speed mode.

Figure 27.28 Keyboard Buffer Controller Timing



Note: * S, P, and Sr indicate the following conditions.

- S: Start condition
- P: Stop condition
- Sr: Retransmission start condition

Figure 27.29 I²C Bus Interface Input/Output Timing

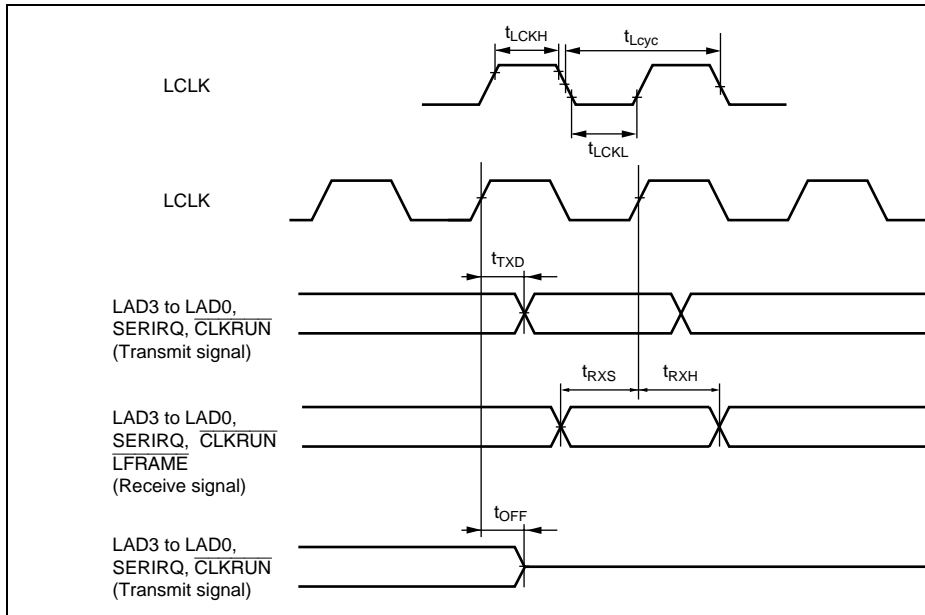


Figure 27.30 Host Interface (LPC) Timing

									output/ input port
	<u>2, 3 (EXPE = 0)</u>								I/O port
Port 2	1	L	T	kept*	kept*	kept*	kept*	A15 to A8	Address
A15 to A8	<u>2, 3 (EXPE = 1)</u>	T							output/ input port
	<u>2, 3 (EXPE = 0)</u>								I/O port
Port 3	1	T	T	T	T	T	T	D15 to D8	
D15 to D8	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>			kept	kept	kept	kept		I/O port
Port 4	1	T	T	kept	kept	kept	kept		I/O port
	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>								
Port 5	1	T	T	kept	kept	kept	kept		I/O port
	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>								
Port 6	1	T	T	kept	kept	kept	kept		I/O port
	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>								
Port 7	1	T	T	T	T	T	T		Input port
	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>								
Port 8	1	T	T	kept	kept	kept	kept		I/O port
	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>								
Port 97	1	T	T	T/kept	T/kept	T/kept	T/kept	WAIT/	WAIT/ I/O port
WAIT	<u>2, 3 (EXPE = 1)</u>								
	<u>2, 3 (EXPE = 0)</u>			kept	kept	kept	kept		I/O port

		2, 3 (EXPE = 0)			kept	kept	kept	kept	I/O port
Ports 92, 91	1	T	T		kept	kept	kept	kept	I/O port
	2, 3 (EXPE = 1)								
	2, 3 (EXPE = 0)								
Port 90	1	T	T	H/kept	H/kept	H/kept	H/kept	LWR/	I/O port
LWR	2, 3 (EXPE = 1)								
	2, 3 (EXPE = 0)			kept	kept	kept	kept	kept	I/O port
Port A	1	T	T	kept*	kept*	kept*	kept*	I/O port	
A23 to A16	2, 3 (EXPE = 1)							A23 to A16/	I/O port
	2, 3 (EXPE = 0)							I/O port	
Port B	1	T	T	T/kept	T/kept	T/kept	T/kept	D7 to D0/	I/O port
D7 to D0	2, 3 (EXPE = 1)								
	2, 3 (EXPE = 0)			kept	kept	kept	kept	kept	I/O port
Ports C to G	1	T	T	kept	kept	kept	kept	I/O port	
(H8S/2160B, H8S/2161B)	2, 3 (EXPE = 1)								
	2, 3 (EXPE = 0)								

Legend:

H: High

L: Low

T: High-impedance state

kept: Input ports are in the high-impedance state (when DDR = 0 and PCR = 1, input p MOSs remain on).

Output ports maintain their previous state.

Depending on the pins, the on-chip peripheral modules may be initialized and the function determined by DDR and DR used.

DDR: Data direction register

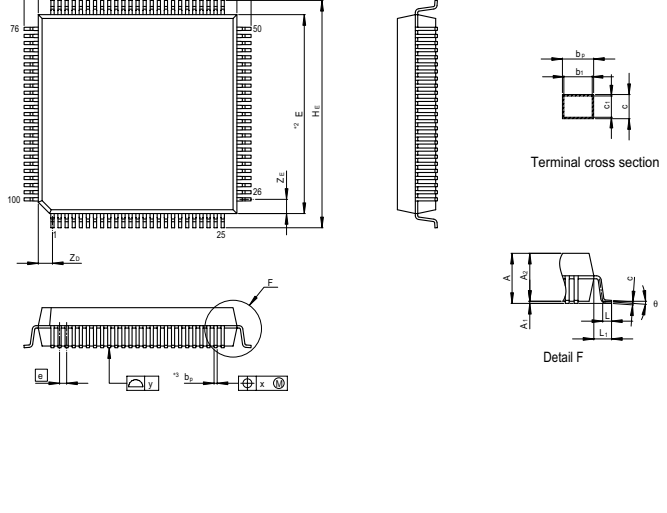
Note: * In the case of address output, the last address accessed is retained.

	(3-V version)		F2141BVTE10	100-pin TQFP
H8S/2140B	Flash memory version (3-V version)	HD64F2140BV	F2140BVFA10	100-pin QFP
			F2140BVTE10	100-pin TQFP
H8S/2145B	Flash memory version (3-V version)	HD64F2145BV	F2145BVFA10	100-pin QFP
			F2145BVTE10	100-pin TQFP
	Flash memory version (5-V version)	HD64F2145B	F2145BFA20	100-pin QFP
			F2145BTE20	100-pin TQFP
H8S/2148B	Flash memory version (3-V version)	HD64F2148BV	F2148BVFA10	100-pin QFP
			F2148BVTE10	100-pin TQFP
	Flash memory version (5-V version)	HD64F2148B	F2148BFA20	100-pin QFP
			F2148BTE20	100-pin TQFP

Legend:

(***): ROM code

Note: * Some products above are in the developing or planning stage. Please contact your local sales agency to conform the present state of each product.



Reference Symbol
D
E
A ₂
H _D
H _E
A
A ₁
b _D
b ₁
c
c ₁
θ
θ
x
y
Z _D
Z _E
L
L ₁

Figure C.1 Package Dimensions (FP-100B)

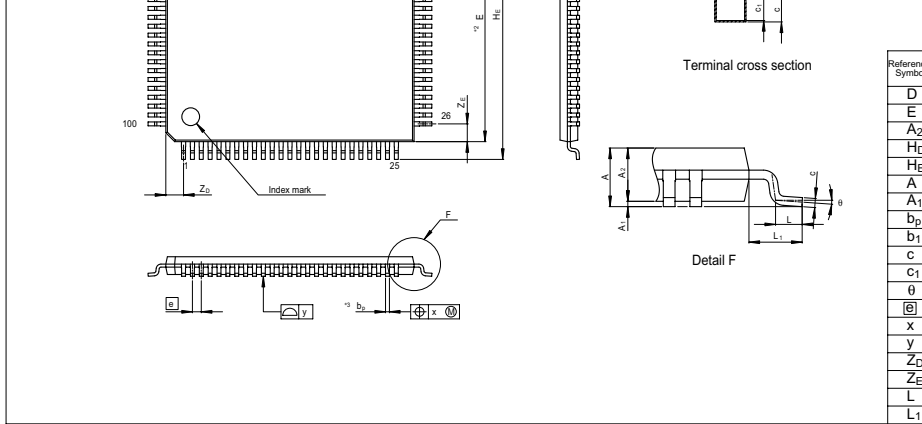


Figure C.2 Package Dimensions (TFP-100B)

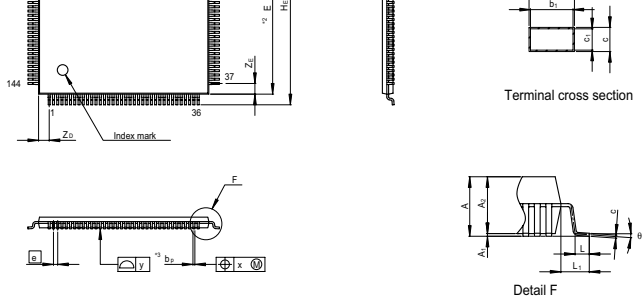


Figure C.3 Package Dimensions (TFP-144)

8-Bit PWM Timer (PWM).....	231
8-Bit Timer (TMR).....	287
A/D Converter	585
A20 Gate.....	521, 565
ABRKCR.....	93, 663, 673, 682, 692
Absolute Address.....	53
Activation by Software	163
ADCR.....	590, 668, 678, 687, 698
ADCSR.....	589, 668, 678, 686, 698
ADDR.....	588, 668, 677, 686, 698
Address Map.....	71
Address Space.....	32
Addressing Modes	51
ADI.....	596
Analog Input Channel.....	588
Arithmetic Operations Instructions.....	43
Asynchronous Mode.....	376
BAR.....	94, 664, 673, 682, 692
Bcc.....	48
BCR.....	124, 667, 676, 685, 696
Bit Manipulation Instructions	46
bit rate	369
Block Configuration	608
Block Data Transfer Instructions.....	50
Block Transfer Mode.....	158
Boot Mode	618
Branch Instructions.....	48
Break.....	406
BRR.....	369, 668, 677, 686, 697
Buffered Input Capture Input.....	274

Chain Transfer	
Clamp Waveform Generation	
Clear Timing	
Clock Pulse Generator	
Clocked Synchronous Mode	
CMI.....	
CMIA	
CMIAY	
CMIB	
CMIBY	
Compare-Match Count Mode	
Condition field	
Condition-Code Register (CCR).....	
Conversion Time.....	
CRA	
CRB.....	
Crystal Resonator.....	
D/A Converter.....	
DACNT.....	245, 666, 675, 684, 693
DACR	248, 581, 665, 674, 683, 692
DADR0	580, 669, 678, 687, 696
DADR1	580, 669, 678, 687, 696
DAR.....	
Data Transfer Controller (DTC).....	
Data Transfer Instructions.....	
DDCSWI.....	
DDCSWR	436, 663, 672, 681, 690
Direct Transitions.....	
DTC Vector Table.....	
DT CER	150, 663, 672, 681, 690

erasing units	608
ERI.....	405
ERRI.....	575
Error Protection	628
Exception Handling	81
Exception Vector Table	82
Extended Control Register (EXR).....	35
External Trigger.....	595
flash memory	603
FLMCR1.....	611, 664, 674, 682, 692
FLMCR2.....	613, 664, 674, 682, 692
Formatless.....	442
FOV	279
framing error.....	383
FRC.....	262, 664, 674, 683, 693
General Registers.....	34
Hardware Protection	628
Hardware Standby Mode	652
HICR.....	514, 669, 678, 687, 698
HICR0.....	533, 661, 672, 681, 690
HICR1.....	533, 661, 672, 681, 690
HICR2.....	540, 661, 672, 681, 690
HICR3.....	540, 661, 672, 681, 690
HIRQ	526
HISEL.....	561, 661, 672, 681, 690
Host Interface LPC Interface (LPC)	529
Host Interface X-Bus Interface (XBS)....	509
HSYNCO Output.....	341

ICR.....	92, 262, 665, 665, 671, 681
ICSR.....	431, 668, 671, 671, 671
ICXR.....	438, 662, 671, 671
Idle Cycle	
IDR.....	517, 661, 671, 671
IER.....	96, 667, 671, 671
IHI signal divided waveform	
IICI.....	
Immediate	
Increment Timing.....	
Input Capture Input.....	
Instruction Set	
Interrupt Control Modes.....	
Interrupt Controller	
Interrupt Exception Handling.....	
Interrupt Exception Handling Ve	
Interrupt Mask Bit.....	
Interval Timer Mode	
IrDA Operation	
ISCR.....	95, 663, 671, 671
ISR	96, 663, 671, 671
KBBR.....	495, 662, 671, 671
KBCOMP.....	591, 663, 671, 671
KBCR.....	492, 662, 671, 671
Keyboard Buffer Controller.....	
KMIMR	97, 669, 671, 671
KMIMRA.....	97, 669, 671, 671
KMPCR	190, 669, 671, 671

Memory Indirect.....	54
Module Stop Mode	656
MRA	147
MRB	149
MSTPCR.....	645, 664, 674, 683, 692
Multiprocessor Communication Function	387
NMI interrupt.....	100
Noise Canceler.....	472
Normal Mode.....	156, 164
OCI	279
OCR.....	262, 665, 675, 683, 693
OCRDM.....	263
ODR.....	517, 545, 661, 680, 690
ODR1	672
On-Board Programming Modes.....	617
Operation field.....	50
Output Compare Output.....	273
overrun error	383
OVI.....	309
OVIY	309
P1DDR.....	172, 666, 676, 685, 695
P1DR	173, 666, 676, 685, 695
P1PCR	173, 666, 676, 684, 695
P2DDR.....	175, 666, 676, 685, 695
P2DR	176, 666, 676, 685, 695
P2PCR	176, 666, 676, 684, 695
P3DDR.....	179, 666, 676, 685, 695
P3DR	180, 666, 676, 685, 696

P8DDR.....	195, 667, 6
P8DR.....	196, 667, 6
P9DDR.....	200, 667, 6
P9DR.....	201, 667, 6
PADDR	205, 666, 6
PAODR	205, 666, 6
PAPIN	206, 666, 6
parity error	
PBDDR	211, 667, 6
PBODR	212, 667, 6
PBPIN	212, 667, 6
PCDDR	218, 662, 6
PCNOCR.....	221, 660, 6
PCODR	219, 662, 6
PCPIN	220, 662, 6
PCSR.....	238, 664, 6
PDDDR	218, 662, 6
PDNOCR	221, 660, 6
PDODR	219, 662, 6
PDPIN	220, 662, 6
PEDDR	223, 661, 6
PENOCR.....	226, 660, 6
PEODR	224, 661, 6
PEPIN	225, 661, 6
PFDDR.....	223, 662, 6
PFNOCR.....	226, 660, 6
PFODR.....	224, 661, 6
PFPIN.....	225, 661, 6
PGDDR	228, 661, 6
PGNOCR	229, 660, 6
PGODR	228, 661, 6
PGPIN	229, 661, 6

PWDR.....	236, 668, 677, 686, 697	Stack Status.....	66, 667, 67
PWM Decoding.....	329	STCR.....	66, 667, 67
PWOER.....	237, 667, 677, 686, 697	STR.....	545, 661, 67
PWSL.....	234, 668, 677, 686, 697	Subactive Mode.....	
RDR.....	362, 668, 677, 686, 697	Subsleep Mode.....	
Register Direct.....	52	SWDTEND.....	
Register field.....	50	SYSCR.....	65, 667, 67
Register Indirect.....	52	SYSCR2.....	512, 664, 67
Register Indirect with Displacement.....	52	System Control Instructions.....	
Register Indirect with Post-Increment.....	52	TCNT.....	291, 347, 666, 668, 684, 68
Register Indirect with Pre-Decrement.....	53	TCNT Count Timing.....	
Register information.....	153	TCONRI.....	320, 670, 67
Repeat Mode.....	157	TCONRO.....	323, 670, 67
Reset.....	83	TCONRS.....	325, 670, 67
Reset Exception Handling.....	83	TCOR.....	291, 667, 67
RSR.....	362	TCORC.....	300, 669, 67
RXI.....	405	TCR.....	268, 291, 665, 668, 683, 68
SAR.....	418, 668, 677, 686, 697	TCSR.....	265, 294, 348, 664, 667, 675, 676, 683, 684, 685, 69
SARX.....	419, 668, 677, 686, 697	TDR.....	362, 668, 67
SBYCR.....	642, 664, 674, 683, 692	TEI.....	
SCMR.....	368, 668, 677, 686, 697	TICR.....	
SCR.....	364, 668, 677, 686, 697	TICRF.....	300, 669, 67
SEDGR.....	327, 670, 679, 688, 700	TICRR.....	300, 669, 67
Serial Communication Interface (SCI and IrDA).....	359	TIER.....	264, 664, 67
Serial Formats.....	443	Timer Connection.....	
Shift Instructions.....	45	TISR.....	300, 669, 67
Shutdown Function.....	523	TOCR.....	269, 665, 67
Single Mode.....	592	Toggle output.....	
SIRQCR.....	552, 661, 672, 680, 690		



**Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8S/2140B Group**

Publication Date: 1st Edition, March 2002

Rev.3.00, March 21, 2006

Published by: Sales Strategic Planning Div.
Renesas Technology Corp.

Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

©2006. Renesas Technology Corp., All rights reserved. Printed in Japan.



RENESAS SALES OFFICES

<http://www.renesas.com/en/network>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

Renesas Technology America, Inc.

450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.

Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.

7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.

10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan
Tel: <603> 7955-9390, Fax: <603> 7955-9510



H8S/2140B Group Hardware Manual



Renesas Electronics Corporation

1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0300-0300

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [16-bit Microcontrollers - MCU category](#):

Click to view products by [Renesas manufacturer](#):

Other Similar products are found below :

[M30302FCFP#U3](#) [MB90F036APMC-GSE1](#) [MB90F428GCPFR-GSE1](#) [MB96F683RBPMC-GSAE1](#) [R5F10MMGDFB#30](#)
[R5F111PGGFB#30](#) [R5F117BCGNA#20](#) [DF3026XBL25V](#) [DF36014GFTV](#) [DF36014GFXV](#) [DF36034GFPV](#) [R5F11B7EANA#U0](#)
[R5F21172DSP#U0](#) [MB90092PF-G-BNDE1](#) [MB90F335APMC1-G-SPE1](#) [MB90F345CAPFR-GSE1](#) [MB90F568PMCR-GE1](#)
[MB96F395RSAPMC-GSE2](#) [DF36024GFXV](#) [UPD78F1018F1-BA4-A](#) [MB96F018RBPMC-GSE1](#) [MB90F867ASPFR-GE1](#)
[M30290FCHP#U3A](#) [DF2239FA20IV](#) [R5F117BCGFP#30](#) [LC88F58B0AU-SQFPH](#) [MB90F548GPF-GE1](#) [MB90214PF-GT-310-BND-AE1](#)
[MB90F342CESPQC-GSE2](#) [MB90F428GAPF-GSE1](#) [ML62Q504H-NNNTBWBX](#) [S912ZVH128F2VLL](#) [UPD78F1500AGK-GAK-AX](#)
[HD64F3337SF16V](#) [MB90F428GCPF-GSE1](#) [MB90F342ESPMC-G-JNE1](#) [MB90022PF-GS-358E1](#) [MB96F395RWAPMC-GSE2](#)
[MB96395RSAPMC-GS-110E2](#) [MB90F883CSPMC-GE1](#) [S912ZVHY64F1CLL](#) [S912ZVHY64F1VLQ](#) [ST10F280](#) [MB96F338RSAPMCR-](#)
[GK5E2](#) [CY90096PF-G-002-BND-ERE1](#) [ML62Q1569-NNNGAZ0AX](#) [ML62Q1739-NNNGAZ0AX](#) [ML62Q1749-NNNGAZ0AX](#)
[ML62Q1579-NNNGAZ0AX](#) [ML62Q1559-NNNGAZ0AX](#)