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.

User's Manual

16

H8S/2148 Group, H8S/2144 Group, H8S/2148F-ZTAT™, H8S/2147N F-ZTAT™, H8S/2144F-ZTAT™, H8S/2142F-ZTAT™

Hardware Manual

Renesas 16-Bit Single-Chip Microcomputer
H8S Family/H8S/2100 Series

H8S/2148	HD6432148S	H8S/2147N	HD64F2147N
	HD6432148SW		HD64F2147NV
	HD64F2148	H8S/2144	HD6432144S
	HD64F2148V		HD64F2144
	HD64F2148A		HD64F2144V
	HD64F2148AV		HD64F2144A
H8S/2147	HD6432147S		HD64F2144AV
	HD6432147SW	H8S/2143	HD6432143S
	HD64F2147A	H8S/2142	HD6432142
	HD64F2147AV		HD64F2142R
	HD64F2142RV		

Renesas Electronics
www.renesas.com

Rev.4.00 2006.09

1. These materials are intended as a reference to assist our customers in the selection of the R
   Technology Corp. product best suited to the customer's application; they do not convey any
   under any intellectual property rights, or any other rights, belonging to Renesas Technology
   a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of ar
   party's rights, originating in the use of any product data, diagrams, charts, programs, algorith
   circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, progra
   algorithms represents information on products at the time of publication of these materials, a
   subject to change by Renesas Technology Corp. without notice due to product improvement
   other reasons. It is therefore recommended that customers contact Renesas Technology C
   an authorized Renesas Technology Corp. product distributor for the latest product informatio
   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
   from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various
   including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.co
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. R
   Technology Corp. assumes no responsibility for any damage, liability or other loss resulting
   information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a d
   system that is used under circumstances in which human life is potentially at stake. Please
   Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor
   considering the use of a product contained herein for any specific purposes, such as appara
   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
   whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they
   be exported under a license from the Japanese government and cannot be imported into a c
   other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or
   country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the produ
   contained therein.

RENESAS

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
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 throug
chip and a low level is input on the reset pin. During the period where the stat
undefined, the register settings and the output state of each pin are also undefi
your system so that it does not malfunction because of processing while it is in
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
may have been be allocated to these addresses. Do not access these registers; t
operation is not guaranteed if they are accessed.

Programs based on the high-level language C can also be run efficiently.

Single-power-supply flash memory (F-ZTAT™*) and mask ROM versions are availa
providing a quick and flexible response to conditions from ramp-up through full-scale
production, even for applications with frequently changing specifications.

On-chip peripheral functions include a 16-bit free-running timer (FRT), 8-bit timer (T
watchdog timer (WDT), two PWM timers (PWM and PWMX), a serial communicatio
(SCI, IrDA), PS/2-compatible keyboard buffer controller, host interface (HIF), D/A co
(DAC), A/D converter (ADC), and I/O ports. An I^2C bus interface (IIC) can also be in
as an option.

An on-chip data transfer controller (DTC) is also provided, enabling high-speed data t
without CPU intervention.

The H8S/2148 Group has all the above on-chip supporting functions, and can also be
with an IIC module as an option. The H8S/2144 Group comprises reduced-function ve
fewer TMR channels, and no PWM, keyboard buffer controller, HIF, IIC, or DTC mo
the H8S/2147N with fewer TMR channels, no DTC and some other functions.

Use of the H8S/2148 Group, H8S/2144 Group, H8S/2147N enables compact, high-pe
systems to be implemented easily. The comprehensive PC-related interface functions
matrix key-scan functions are ideal for applications such as notebook PC keyboard co
intelligent battery and power supply control, while the various timer functions and the
interconnectability (timer connection), plus the interlinked operation of the I^2C bus int
data transfer controller (DTC), in particular, make these devices ideal for use in PC m
addition, the combination of F-ZTAT™* and reduced-function versions is ideal for ap
such as CD-ROM drive units in which on-chip program memory is essential to meet p
requirements, product start-up times are short, and program modifications may be nec
end-product assembly.

Product names		H8S/2148, H8S/2147	H8S/2147N	H8S/21 H8S/21 H8S/21
Bus controller (BSC)		Available (16 bits)	Available (16 bits)	Availab
Data transfer controller (DTC)		Available	—	—
8-bit PWM timer (PWM)		×16	×16	—
14-bit PWM timer (PWMX)		×2	×2	×2
16-bit free-running timer (FRT)		×1	×1	×1
8-bit timer (TMR)		×4	×3	×3
Timer connection		Available	—	—
Watchdog timer (WDT)		×2	×2	×2
Serial communication interface (SCI)		×3	×3	×3
I^2C bus interface (IIC)		×2 (option)	×2 (option)	—
Keyboard buffer controller (PS/2 compatible)		×3	×3	—
Host interface (HIF)		×4	×4	—
D/A converter		×2	×2	×2
A/D converter	Analog inputs	×8	×8	×8
	Expansion A/D inputs	×16	×16	×16

(Before) H8S/2144 Series → (After) H8S/2144

1.1 Overview

Table 1.1 Overview

4 Host interface specification in table 1.1 amended

- 8-bit host interface (ISA) port

6 Product lineup specification in table 1.1 amended

Group	Product Code[*2] Mask ROM Versions	F-ZTAT Versions	ROM/RAM (Bytes)	P T
H8S/2148	HD6432148S	HD64F2148 HD64F2148V[*2]	128 k/4 k	
	HD6432148SW[*1]	HD64F2148A HD64F2148AV[*2]		
	HD6432147S	HD64F2147A	64 k/2 k	
	HD6432147SW[*1]	HD64F2147AV[*2]		
H8S/2147N	—	HD64F2147N HD64F2147NV[*2]	64 k/2 k	
H8S/2144	HD6432144S	HD64F2144 HD64F2144V[*2]	128 k/4 k	
		HD64F2144A HD64F2144AV[*2]		
	HD6432143S	—	96 k/4 k	
	HD6432142	HD64F2142R HD64F2142RV[*2]	64 k/2 k	

Notes: 1. W indicates the I^2C bus option.
2. V indicates the 3-V version. Please refer to appendix Code Lineup.

1.2 Internal Block Diagram

Figure 1.1 (a) Internal Block Diagram of H8S/2148 Group

7 Figure 1.1 (a) amended

(Before) STBY → (After) $\overline{\text{STBY}}$

Figure 1.1 (b) Internal Block Diagram of H8S/2147N

8 Figure 1.1 (b) amended

(Before) IIC × 2ch → (After) IIC × 2ch (option)

RENESAS

Functions in Each Operating Mode	21	Modes 2 and 3 in single chip modes of pin 95 amend (Before) P82 → (After) P82/HIFSD
1.3.3 Pin Functions Table 1.3 Pin Functions	30	Table 1.3 amended

		Pin No.		
Type	Symbol	FP-100B TFP-100B	I/O	Name and Function
Host interface (HIF)	HIRQ11 HIRQ1 HIRQ12 HIRQ3 HIRQ4	52 53 54 91 90	Output	**Host interrupt 11, 1, 12, 3, and 4** interrupt requests to the host.

		Pin No.		
Type	Symbol	FP-100B TFP-100B	I/O	Name and Function
I/O ports	PA7 to PA0	10, 11, 20, 21, 30, 31, 47, 48	Input/ output	**Port A:** Eight input/output pins. Th of each pin can be selected in the direction register (PADDR). These MOS input pull-ups. These are the pins. [H8S/2148 Group and H8S/2

2.6.1 Overview Table 2.1 Instruction Classification	52	Instruction in arithmetic operations amended (Before) EG → (After) NEG
3.2.4 Serial Timer Control Register (STCR)	89	Bit 3 bit table amended

Bit 3	
FLSHE	Description
0	Addresses H'(FF)FF80 to H'(FF)FF87 are used for power-down mod and supporting module control register access
1	Addresses H'(FF)FF80 to H'(FF)FF87 are used for flash memory cor access (F-ZTAT version only)

4.5 Stack Status after Exception Handling Figure 4.5 (2) Stack Status after Exception Handling (Advanced Mode)	111	Note * deleted from figure 4.5 (2)

8.1 Overview 213 Table 8.1 amended

Table 8.1 H8S/2148 Group Port Functions

Port	Description	Pins	Expanded Modes	
			Mode 1	Mode 2, Mode 3 (EXPE = 1)
Port A	• 8-bit I/O port	PA7/A23/KIN15/ CIN15/PS2CD PA6/A22/KIN14/ CIN14/PS2CC PA5/A21/KIN13/ CIN13/PS2BD PA4/A20/KIN12/ CIN12/PS2BC PA3/A19/KIN11/ CIN11/PS2AD PA2/A18/KIN10/ CIN10/PS2AC PA1/A17/KIN9/ CIN9 PA0/A16/KIN8/ CIN8	I/O port also functioning as key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also functioning as address output (A23 to A16), key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/ output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)

Table 8.2 H8S/2147N Port Functions 216 Table 8.2 amended

Port	Description	Pins	Expanded Modes	
			Mode 1	Mode 2, Mode 3 (EXPE = 1)
Port A	• 8-bit I/O port	PA7/A23/KIN15/ CIN15/PS2CD PA6/A22/KIN14/ CIN14/PS2CC PA5/A21/KIN13/ CIN13/PS2BD PA4/A20/KIN12/ CIN12/PS2BC PA3/A19/KIN11/ CIN11/PS2AD PA2/A18/KIN10/ CIN10/PS2AC PA1/A17/KIN9/ CIN9 PA0/A16/KIN8/ CIN8	I/O port also functioning as key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also functioning as address output (A23 to A16), key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/ output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)

8.7.2 Register Configuration Table 8.14 Port 6 Registers	247	Table 8.14 amended (Before) System control register → (After) System co register 2
8.9.3 Pin Functions Table 8.19 Port 8 Pin Functions	258	P81/GA20/CS2 selection method and pin function de amended

Pin | **Selection Method and Pin Functions**

P81/GA20/$\overline{CS2}$

The pin function is switched as shown below according to the operating mode, bit CS2E in SYSCR, bit FGA20E in HICR of P81DDR.

Operating mode	Not slave mode		Slave mode		
FGA20E	—		0		
CS2E	—		0		1
P81DDR	0	1	0	1	—
Pin function	P81 input pin	P81 output pin	P81 input pin	P81 output pin	$\overline{CS2}$ input pin

This pin should be used as the GA20 output pin or $\overline{CS2}$ input 2 or 3 (EXPE = 0).

This pin can always be used as the KIN9 or CIN9 input pin.

PA0/A16/$\overline{\text{KIN8}}$/ CIN8

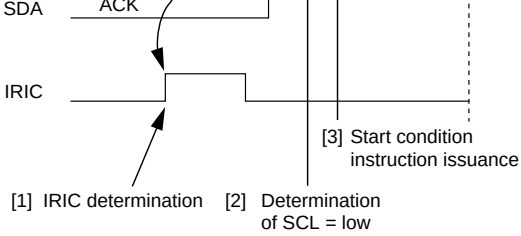
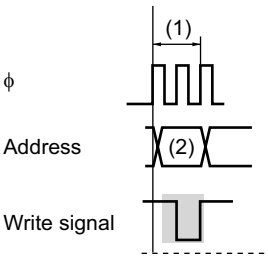
The pin function is switched as shown below according to the co
operating mode, the IOSE bit in SYSCR, and bit PA0DDR.

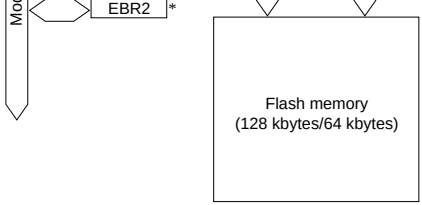
Operating mode	Modes 1, 2 (EXPE = 0), 3		Mode 2 (EXPE	
PA0DDR	0	1	0	
IOSE	—	—	—	0
Pin function	PA0 input pin	PA0 output pin	PA0 input pin	A16 output pin
	$\overline{\text{KIN8}}$ input pin, CIN8 input pin			

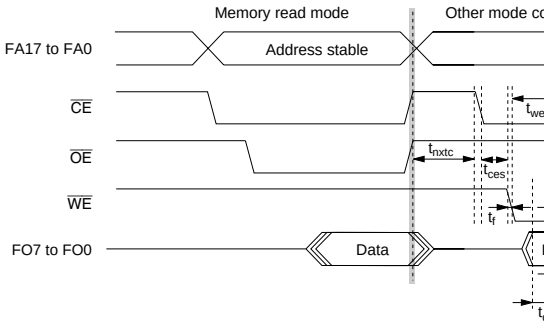
This pin can always be used as the $\overline{\text{KIN8}}$ or CIN8 input pin.

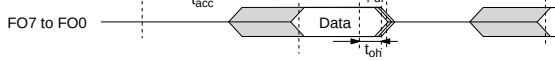
9.3.1 Correspondence between PWM Data Register Contents and Output Waveform Table 9.4 Duty Cycle of Basic Pulse	289	Description of upper 6 bits changed to of upper 4 b
10.3 Bus Master Interface	299	Description amended ... Example 2: Read DADRA `MOV.W @DADRA, R0` ; Transfer contents of DADI
10.4 Operation Table 10.4 Settings and Operation (Examples when ϕ = 10 MHz)	303	Table 10.4 amended

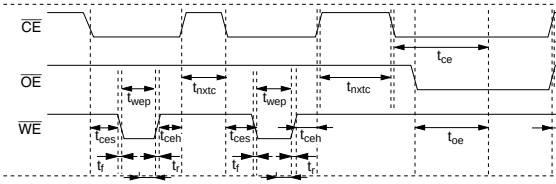
CKS	Resolution T (µs)	CFS	Base Cycle (µs)	Conversion Cycle (µs)	T_L (if OS = 0) T_H (if OS = 1)	Fixed DADR Precision (Bits)	B 3
0	0.1	0	6.4	1638.4	1. Always low (or high) level output (DADR = H'0001 to H'03FD)	14	
		1	25.6	1638.4	1. Always low (or high) level output (DADR = H'0003 to H'00FF)	14	
1	0.2	0	12.8	3276.8	1. Always low (or high) level output (DADR = H'0001 to H'03FD)	14	
		1	51.2	3276.8	1. Always low (or high) level output (DADR = H'0003 to H'00FF)	14	

Retransmission		
	550 to 557	Description added • Notes on WAIT Function • Notes on ICDR Reads and ICCR Access in Slave Mode • Notes on TRS Bit Setting in Slave Mode • Notes on Arbitration Lost in Master Mode • Notes on Interrupt Occurrence after ACKB Recepti • Notes on TRS Bit Setting and ICDR Register Acce
20.4.3 Input Sampling and A/D Conversion Time Figure 20.5 A/D Conversion Time	628	Figure 20.5 amended 



22.5.3 Erase Block Registers 1 and 2 (EBR1, EBR2)	653	Bit figure amended Read/Write description of bits 7 to 2 (Before) —*2 –
22.10.1 Programmer Mode Setting	671	Note amended In programmer mode, ... Renesas Technology micr device types with 128-kbyte*1*3 or 64-kbyte*2*3 on- memory. ... Note: 3. Use products other than the A-mask versi H8S/2148, H8S/2147N, H8S/2144, and H8S/2142
22.10.4 Memory Read Mode Figure 22.17 Timing Waveforms when Entering Another Mode from Memory Read Mode	675	Figure 22.17 amended 

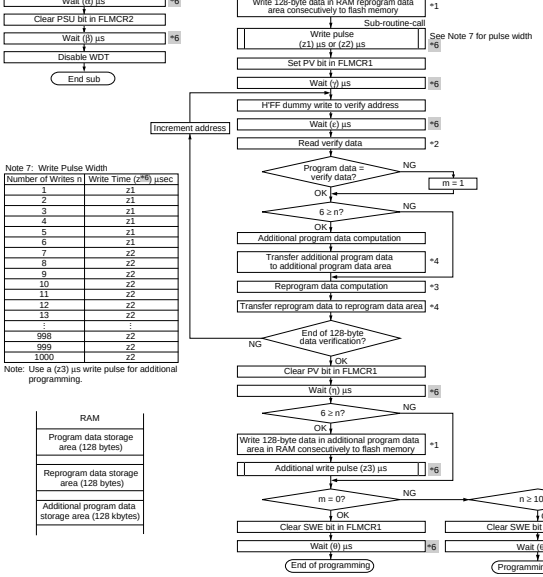


22.10.7 Status Read Mode Figure 22.22 Status Read Mode Timing Waveforms	681	Figure 22.22 amended 
23.5.3 Erase Block Registers 1 and 2 (EBR1, EBR2)	699	Bit figure amended Read/Write description of bits 7 to 2 (Before) —*2 → Bit 1 0 EBR1 EB9/—*2 EB8/—*2 Initial value 0 0 Read/Write R/W*1 *2 R/W*1 *2 Note 2 amended Note: 2. Bits EB8 and EB9 are not present in the 64 versions; they must not be set to1.
Table 23.5 Flash Memory Erase Blocks	700	64-kbyte description added to table 23.5

Block (Size)

128-kbyte Version	64-kbyte Version	Address
EB0 (1 kbyte)	EB0 (1 kbyte)	H'(00)0000 to H'(00)03FF
EB1 (1 kbyte)	EB1 (1 kbyte)	H'(00)0400 to H'(00)07FF
EB2 (1 kbyte)	EB2 (1 kbyte)	H'(00)0800 to H'(00)0BFF
EB3 (1 kbytes)	EB3 (1 kbytes)	H'(00)0C00 to H'(00)0FFF
EB4 (28 kbytes)	EB4 (28 kbytes)	H'(00)1000 to H'(00)7FFF
EB5 (16 kbytes)	EB5 (16 kbytes)	H'(00)8000 to H'(00)BFFF
EB6 (8 kbytes)	EB6 (8 kbytes)	H'(00)C000 to H'(00)DFFF
EB7 (8 kbytes)	EB7 (8 kbytes)	H'00E000 to H'00FFFF
EB8 (32 kbytes)	—	H'010000 to H'017FFF
EB9 (32 kbytes)	—	H'018000 to H'01FFFF

RENESAS

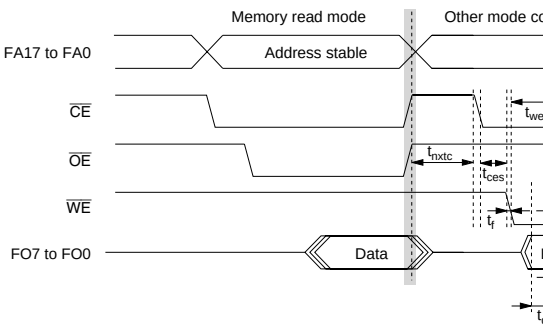


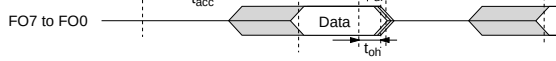
23.10.4 Memory Read Mode

721

Figure 23.17 Timing Waveforms when Entering Another Mode from Memory Read Mode

Figure 23.17 amended



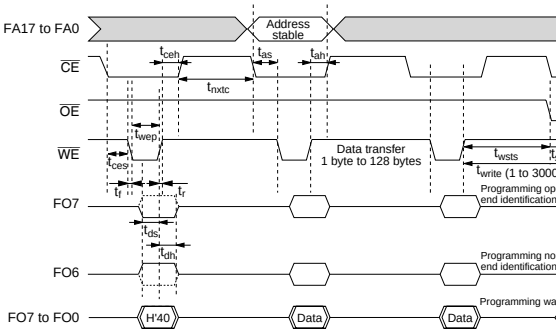


23.10.5 Auto-Program Mode

Figure 23.20 Auto-Program Mode Timing Waveforms

724

Figure 23.20 amended

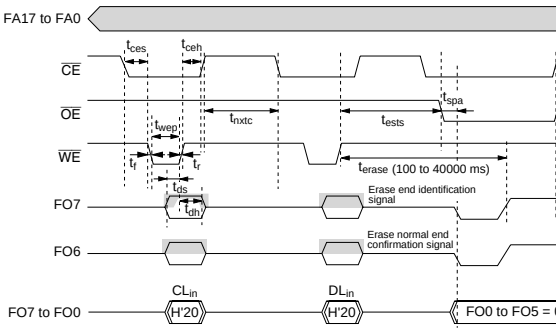


23.10.6 Auto-Erase Mode

Figure 23.21 Auto-Erase Mode Timing Waveforms

725

Figure 23.21 amended



RENESAS

24.7 Subclock Input Circuit	740	Note on Subclock Usage Description added
25.12 Usage Notes	764	Section 25.12 added
26.2.6 Flash Memory Characteristics Table 26.15 Flash Memory Characteristics (Programming/erasing operating range)	799	Table 26.15 amended

Item	Symbol	Min	Typ	Max	U
Reprogramming count	N_{WEC}	100*8	10000*8	—	Ti
Data retention time*10	t_{DRP}	10	—	—	Y
Programming Wait time after SWE-bit setting*1	x	10	—	—	µs

	800 Notes 8 to 10 added

Notes: 8. Minimum number of times for which all characteristics are guaranteed after rewriting (Guar is 1 to minimum value).

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

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

26.3.3 AC Characteristics Table 26.20 Clock Timing	819 Table 26.20 amended

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	Symbol	Min	Max	Min	Max	Min	Max	Un
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	10	—	20	—	ms
Oscillation settling time in software standby (crystal)	t_{OSC2}	8	—	8	—	8	—	ms

Table 26.25 I^2C Bus Timing	829 Table 26.25 amended

		Ratings				
Item	Symbol	Min	Typ	Max	Unit	Test Conditio
SCL, SDA input fall time	t_{sf}	—	—	300	ns	
SCL, SDA output fall time	t_{of}	20 + 0.1 Cb	—	250	ns	
SCL, SDA input spike pulse elimination time	t_{SP}	—	—	1	t_{cyc}	

RENESAS

Characteristics Table 26.29 Flash Memory Characteristics (Programming/erasing operating range)		Item amended (Before) Wait time after dummy write Wait time after H'FF dummy write Symbol of wait time after SWE-bit clear (Before) Θ – **Item** **Symbol** **Min** **Typ** **Max** **Uni** Reprogramming count N_{WEC} 100*9 10000*9 — Tim Data retention time*10 t_{DRP} 10 — — Yea Programming Wait time after SWE-bit setting*1 x 1 — — µs
	834	Notes 8 to 10 added Notes: 8. Minimum number of times for which all characteristics are guaranteed after rewriting (Guara is 1 to minimum value). 9. Reference value for 25°C (as a guideline, rewritin normally function up to this value). 10. Data retention characteristic when rewriting is p within the specification range, including the minimum
26.4.3 AC Characteristics Table 26.35 Control Signal Timing	849	Unit of t_{NMIH} amended (Before) — → (After) ns
Table 26.37 Timing of On-Chip Supporting Modules (1)	854, 855	Units of t_{PRS}, t_{PRH}, t_{FTIS}, t_{FTCS}, t_{TMRS}, t_{TMCS} amended (Before) — → (After) ns Units of t_{FTCWL}, t_{TMCWL}, synchronous t_{Scyc} amended (Before) — → (After) t_{cyc} Unit of t_{SCKf} amended (Before) 1.5 → (After) t_{cyc}
26.4.4 A/D Conversion Characteristics Table 26.41 A/D Conversion Characteristics (CIN15 to CIN0 Input: 134/266-State Conversion)	860	Table condition amended Table condition A (Before) ..., Ta = –20 to +75°C (re specifications), Ta = –40 to +85°C (wide-range spec → (After) ... Ta = –20 to +75°C

RENESAS

		is 1 to minimum value). 9. Reference value for 25°C (as a guideline, rewrit normally function up to this value). 10. Data retention characteristic when rewriting is within the specification range, including the minimu
26.5.3 AC Characteristics Table 26.49 Control Signal Timing	877	Unit of t_{NMIH} amended (Before) (blank) → (After) ns
26.5.6 Flash Memory Characteristics Table 26.55 Flash Memory Characteristics (Programming/erasing operating range)	885	Table 26.55 amended **Item** **Symbol** **Min** **Typ** **Max** **U** Reprogramming count N_{WEC} 100*8 10000*9 — T Data retention time*10 t_{DRP} 10 — — Y Programming Wait time after SWE-bit setting*1 x 10 — — µ
	886	Notes 8 to 10 added Notes: 8. Minimum number of times for which all characteristics are guaranteed after rewriting (Guar is 1 to minimum value). 9. Reference value for 25°C (as a guideline, rewrit normally function up to this value). 10. Data retention characteristic when rewriting is within the specification range, including the minimu
26.6.3 AC Characteristics Table 26.61 Control Signal Timing	900	Unit of t_{NMIH} amended (Before) (blank) → (After) ns

RENESAS

characteristics are guaranteed after rewriting (Guara
is 1 to minimum value).

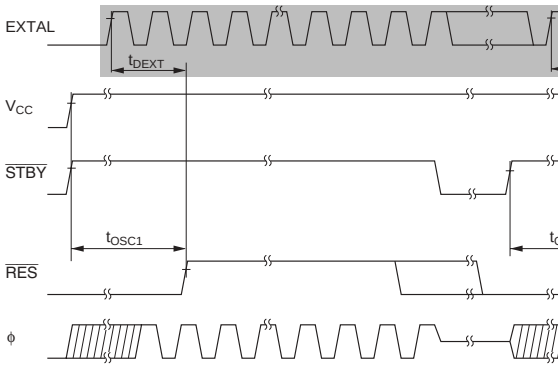
9. Reference value for 25°C (as a guideline, rewritin
normally function up to this value).

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

26.7.2 Clock Timing — 912

Figure 26.6 Oscillation Settling Timing

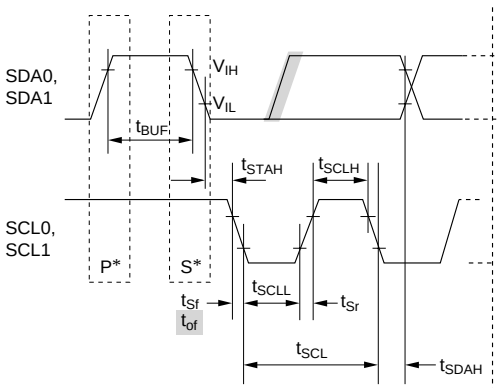
Figure 26.6 amended



26.7.5 Timing of On-Chip Supporting Modules — 924

Figure 26.28 I^2C Bus Interface Input/Output Timing (Option)

Figure 26.28 amended



RENESAS

933 Table A.1 amended

4. Shift Instructions

Mnemonic		Size	Addressing Mode and Instruction Length (Bytes) #xx	Rn	@ERn	@(d,ERn)	@-ERn/@ERn+	@aa	@(d,PC)	@@aa	—	Operation
SHLR	SHLR.B Rd	B		2								0 → C MSB → LSB
	SHLR.B #2,Rd	B		2								
	SHLR.W Rd	W		2								
	SHLR.W #2,Rd	W		2								
	SHLR.L ERd	L		2								
	SHLR.L #2,ERd	L		2								

939 Table A.1 amended

6. Branch Instructions

Mnemonic		Size	Addressing Mode and Instruction Length (Bytes) #xx	Rn	@ERn	@(d,ERn)	@-ERn/@ERn+	@aa	@(d,PC)	@@aa	—	Operation
JMP	JMP @ERn	—			2							PC←ERn
	JMP @aa:24	—						4				PC←aa:24
	JMP @@aa:8	—								2		PC←@aa:8
BSR	BSR d:8	—							2			PC→@-SP,PC←PC+d:8
	BSR d:16	—							4			PC→@-SP,PC←PC+d:16
JSR	JSR @ERn	—			2							PC→@-SP,PC←ERn
	JSR @aa:24	—						4				PC→@-SP,PC←aa:24
	JSR @@aa:8	—								2		PC→@-SP,PC←@aa:8
RTS	RTS	—									2	PC←@SP+

A.2 Instruction Codes

949 Table A.2 amended

Table A.2 Instruction Codes

Instruction	Mnemonic	Size	1st Byte		2nd Byte		3rd Byte		4th Byte	
LDC	LDC @aa:16,CCR	W	0	1	4	0	6	B	0	0
	LDC @aa:16,EXR	W	0	1	4	1	6	B	0	0

RENESAS

	is set (IRQnSCB = IRQnSCA = 0) and IRQn input is high* • When IRQn interrupt exception handling is executed while falling, risin or both-edge detection is set (IRQnSCB = 1 or IRQnSCA = 1)*
	Note: * When a product, in which a DTC is incorpora used in the following settings, the corresponding flag automatically cleared even when exception handling clear condition, is executed and the bit is held at 1. (1) When DTCEA3 is set to 1 (ADI is set to an interru IRQ4F flag is not automatically cleared. (2) When DTCEA2 is set to 1 (ICIA is set to an interr source), IRQ5F flag is not automatically cleared. (3) When DTCEA1 is set to 1 (ICIB is set to an interr source), IRQ6F flag is not automatically cleared. (4) When DTCEA0 is set to 1 (OCIA is set to an inte source), IRQ7F flag is not automatically cleared. When activation interrupt sources of DTC and IRQ in are used with the above combinations, clear the inte by software in the interrupt handling routine of the corresponding IRQ.
1019	ABRKCR—H'FEF4 Interrupt Controller Read/Write description amended Bit 7 (Before) R/W → (After) R
1021	FLMCR1—H'FF80 Flash Memory Initial value description amended Bit 7 (Before) — → (After) 1

Bit	7	6	5	4	3	2
EBR2	EB7	EB6	EB5	EB4	EB3	EB2
Initial value	0	0	0	0	0	0
Read/Write	R/W*1	R/W	R/W	R/W	R/W	R/W

1030 ICCR1—H'FF88 IIC1

ICCR0—H'FFD8 IIC0

Figure amended

I²C bus interface enable

0	I²C bus interface module disabled, with SCL and SDA signal pins set to port function SAR and SARX can be accessed
1	I²C bus interface module enabled for transfer operations (pins SCL and SDA are driving the bus) ICMR and ICDR can be accessed

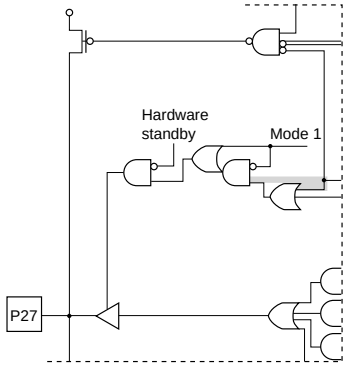
1059 SYSCR—H'FFC4 System

Figure amended

IOS enable

0	The $\overline{AS}$/$\overline{IOS}$ pin functions as the address strobe pin (Low output when accessing an external area)
1	The $\overline{AS}$/$\overline{IOS}$ pin functions as the I/O strobe pin (Low output when accessing a specified address from H'(FF)F000

Note: * In the H8S/2148 F-ZTAT A-mask version and H8S/2147 F-ZTA
the address range is from H'(FF)F000 to H'(FF)F7FF.

		Slave R/W description amended Bit 0 (Before) R → (After) R/(W)
C.2 Port 2 Block Diagrams Figure C.4 Port 2 Block Diagram (Pin P27)	1089	Figure C.4 amended 
Appendix F Product Code Lineup Table F.1 H8S/2148 Group and H8S/2144 Group Product Code Lineup	1128	Package code in table F.1 amended HD64F2144ATE20 (Before) FP-100B → (After) TFP HD64F2144AVFA10 (Before) TFP-100B → (After) F
Appendix G Package Dimensions Figure G.1 Package Dimensions (FP-100B)	1129	Figure G.1 replaced
Figure G.2 Package Dimensions (TFP-100B)	1130	Figure G.2 replaced

RENESAS

RENESAS

The H8S/2000 CPU has an internal 32-bit architecture, is provided with sixteen 16-bit registers and a concise, optimized instruction set designed for high-speed operation, a address a 16-Mbyte linear address space. The instruction set is upward-compatible wi and H8/300H CPU instructions at the object-code level, facilitating migration from th H8/300L, or H8/300H Series.

On-chip supporting modules required for system configuration include a data transfer (DTC) bus master, ROM and RAM, a 16-bit free-running timer module (FRT), 8-bit t (TMR), watchdog timer module (WDT), two PWM timers (PWM and PWMX), serial communication interface (SCI), PS/2-compatible keyboard buffer controller, host inte D/A converter (DAC), A/D converter (ADC), and I/O ports. An I^2C bus interface (IIC incorporated as an option.

The on-chip ROM is either flash memory (F-ZTAT™*) or mask ROM, with a capacit or 64 kbytes. ROM is connected to the CPU via a 16-bit data bus, enabling both byte a data to be accessed in one state. Instruction fetching has been speeded up, and process increased.

Three operating modes, modes 1 to 3, are provided, and there is a choice of address sp single-chip mode or externally expanded modes.

The features of this LSI are shown in table 1.1.

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

- High-speed arithmetic operations
  - 8/16/32-bit register-register add/subtract: 50 ns (20-MHz
  - 16 × 16-bit register-register multiply: 1000 ns (20-MHz op
  - 32 ÷ 16-bit register-register divide: 1000 ns (20-MHz ope
- Instruction set suitable for high-speed operation
  - Sixty-five basic instructions
  - 8/16/32-bit transfer/arithmetic and logic instructions
  - Unsigned/signed multiply and divide instructions
  - Powerful bit-manipulation instructions
- Two CPU operating modes
  - Normal mode: 64-kbyte address space
  - Advanced mode: 16-Mbyte address space

Operating modes

- Three MCU operating modes

Mode	CPU Operating Mode	Description	On-Chip ROM	External Initial Value
1	Normal	Expanded mode with on-chip ROM disabled	Disabled	8 bits
2	Advanced	Single-chip mode	Enabled	None
		Expanded mode with on-chip ROM enabled	Enabled	8 bits
3	Normal	Single-chip mode	Enabled	None
		Expanded mode with on-chip ROM enabled	Enabled	8 bits

	• Transfer possible in repeat mode, block transfer mode, etc. • Request can be sent to CPU for interrupt that activated DT
16-bit free-running timer module (FRT: 1 channel)	• One 16-bit free-running counter (also usable for external ev counting) • Two output compare outputs • Four input capture inputs (with buffer operation capability)
8-bit timer module (2 channels: TMR0, TMR1)	Each channel has: • One 8-bit up-counter (also usable for external event counti • Two timer constant registers • The two channels can be connected
Timer connection and 8-bit timer module (TMR) (2 channels: TMRX, TMRY) (Timer connection and TMRX provided in H8S/2148 Group)	Input/output and FRT, TMR1, TMRX, TMRY can be interconne • Measurement of input signal or frequency-divided waveforn width and cycle (FRT, TMR1) • Output of waveform obtained by modification of input signa TMR1) • Determination of input signal duty cycle (TMRX) • Output of waveform synchronized with input signal (FRT, T TMRY) • Automatic generation of cyclical waveform (FRT, TMRY)
Watchdog timer module (WDT: 2 channels)	• Watchdog timer or interval timer function selectable • Subclock operation capability (channel 1 only)
8-bit PWM timer (PWM) (H8S/2148 Group and H8S/2147N)	• Up to 16 outputs • Pulse duty cycle settable from 0 to 100% • Resolution: 1/256 • 1.25 MHz maximum carrier frequency (20-MHz operation)

SCI with IrDA: 1 channel (SCI2)	• Asynchronous mode or synchronous mode selectable • Multiprocessor communication function • Compatible with IrDA specification version 1.0 • TxD and RxD encoding/decoding in IrDA format
Keyboard buffer controller (PS2: 3 channels) (H8S/2148 Group, H8S/2147N)	• Compatible with PS/2 interface • Direct manipulation of transmission output by software • Receive data input to 8-bit shift register • Data receive completed interrupt, parity error detection, stop monitoring
Host interface (HIF) (H8S/2148 Group, H8S/2147N)	• 8-bit host interface (ISA) port • Five host interrupt requests (HIRQ11, HIRQ1, HIRQ12, HIR HIRQ4) • Normal and fast A20 gate output • Four register sets (each comprising two data registers and tw registers)
Keyboard controller	• Matrix keyboard control using keyboard scan with wakeup in sense port configuration
A/D converter	• Resolution: 10 bits • Input: — 8 channels (dedicated analog pins) — 16 channels (same pins as keyboard sense port) • High-speed conversion: 6.7 µs minimum conversion time (20 operation) • Single or scan mode selectable • Sample-and-hold function • A/D conversion can be activated by external trigger or timer

Memory	• Flash memory or mask ROM • High-speed static RAM

Product Name	ROM	RAM
H8S/2144, H8S/2148	128 kbytes	4 kbytes
H8S/2143	96 kbytes	4 kbytes
H8S/2142, H8S/2147, H8S/2147N	64 kbytes	2 kbytes

Interrupt controller	• Nine external interrupt pins (NMI, $\overline{IRQ0}$ to $\overline{IRQ7}$) • 44 internal interrupt sources • Three priority levels settable
Power-down state	• Medium-speed mode • Sleep mode • Module stop mode • Software standby mode • Hardware standby mode • Subclock operation
Clock pulse generator	• Built-in duty correction circuit
Packages	• 100-pin plastic QFP (FP-100B) • 100-pin plastic TQFP (TFP-100B)
I^2C bus interface (IIC: 2 channels) (option in H8S/2148 Group and H8S/2147N)	• Conforms to Philips I^2C bus interface standard • Single master mode/slave mode • Arbitration lost condition can be identified • Supports two slave addresses

	HD6432147S	HD64F2147A	64 k/2 k
	HD6432147SW*1	HD64F2147AV*2	
H8S/2147N	—	HD64F2147N HD64F2147NV*2	64 k/2 k
H8S/2144	HD6432144S	HD64F2144 HD64F2144V*2	128 k/4 k
		HD64F2144A HD64F2144AV*2	
	HD6432143S	—	96 k/4 k
	HD6432142	HD64F2142R HD64F2142RV*2	64 k/2 k

Notes: 1. W indicates the I^2C bus option.
2. V indicates the 3-V version. Please refer to appendix
Code Lineup.

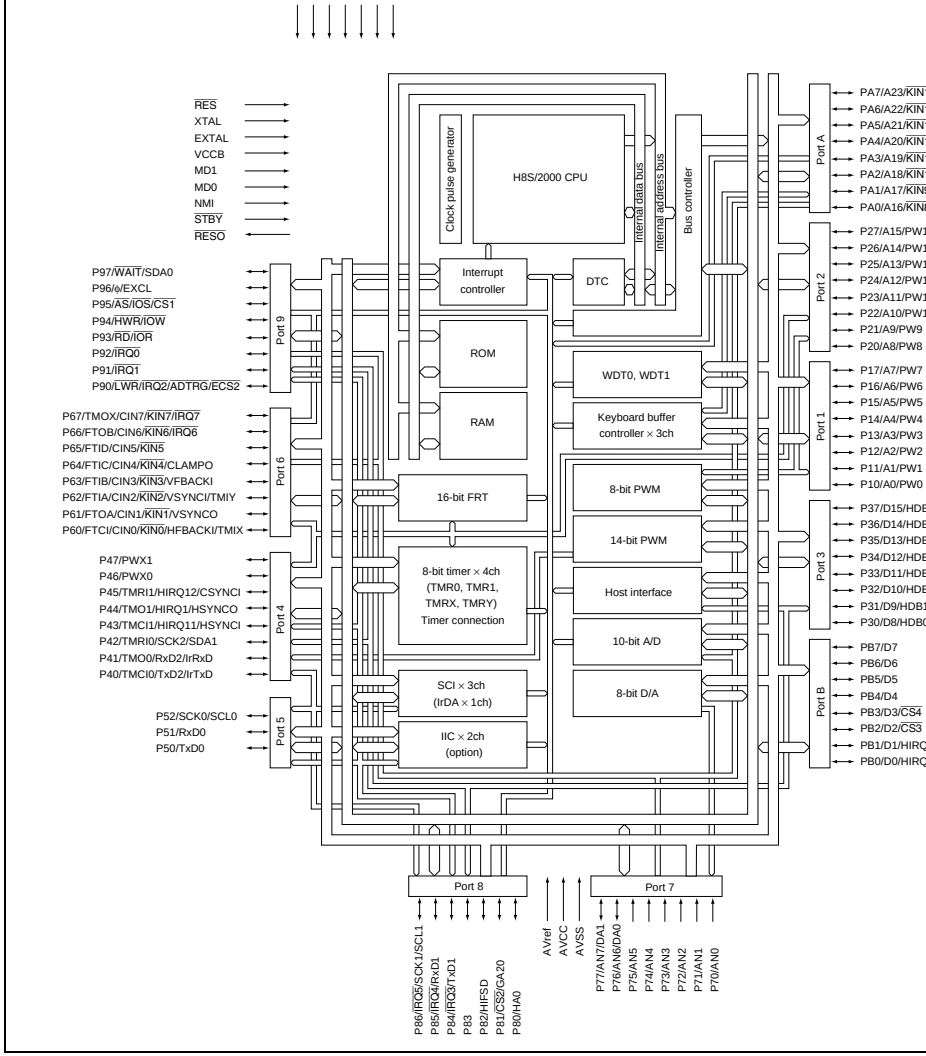


Figure 1.1 (a) Internal Block Diagram of H8S/2148 Group

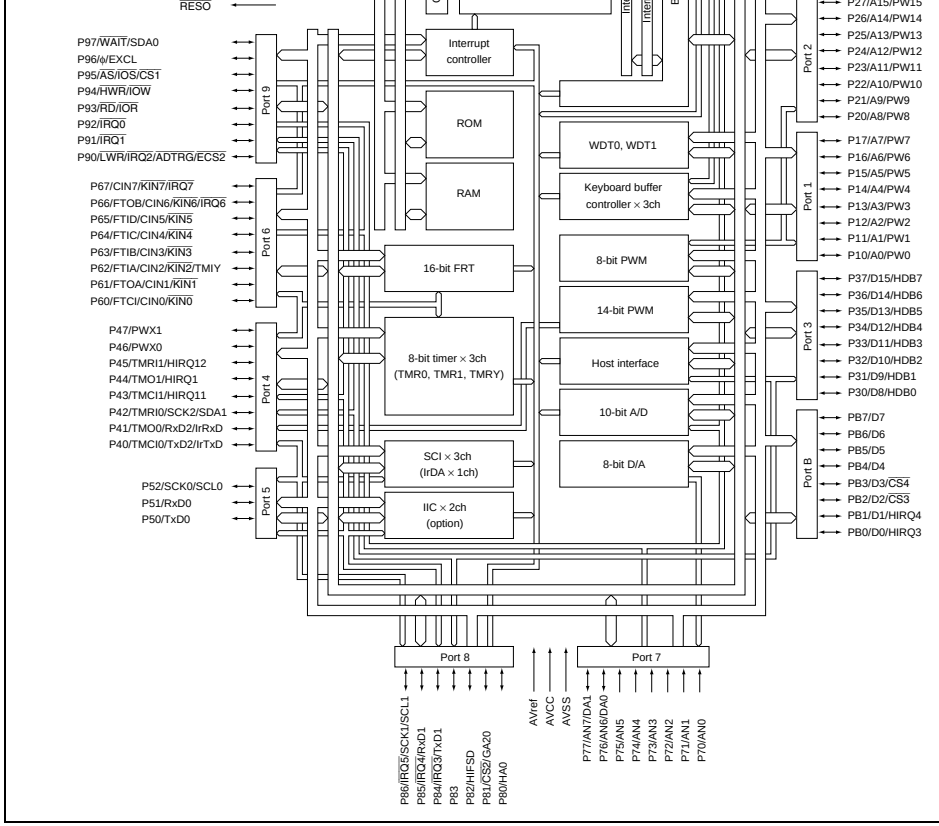


Figure 1.1 (b) Internal Block Diagram of H8S/2147N

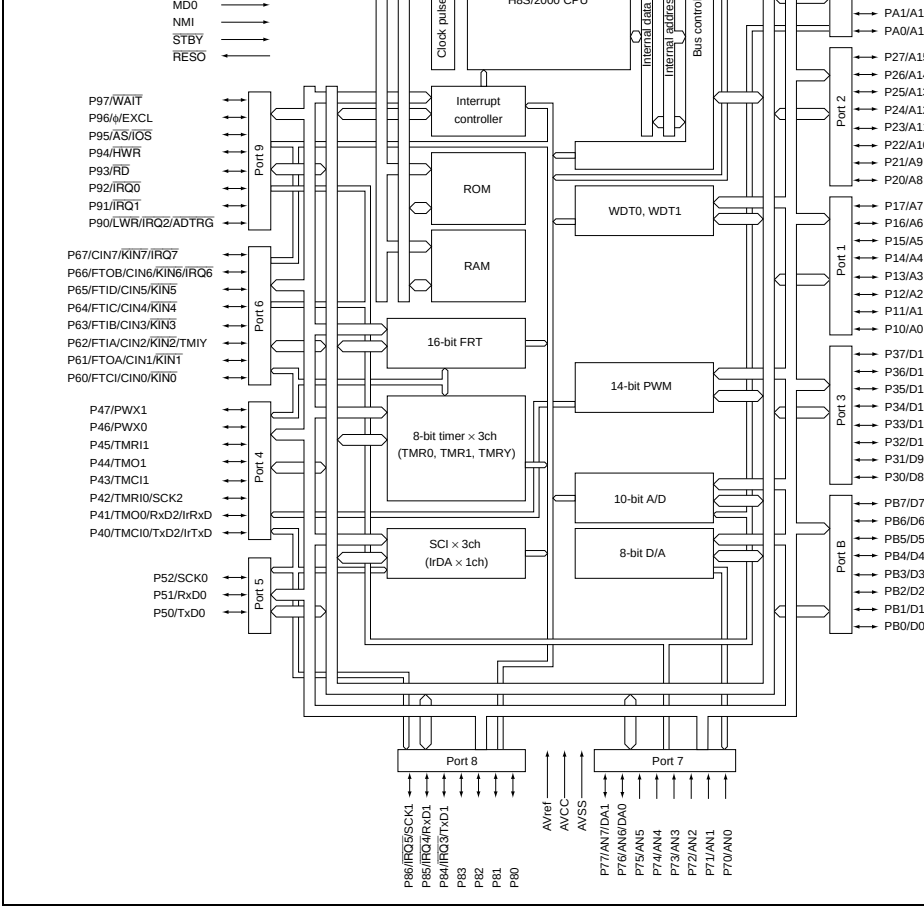


Figure 1.1 (c) Internal Block Diagram of H8S/2144 Group

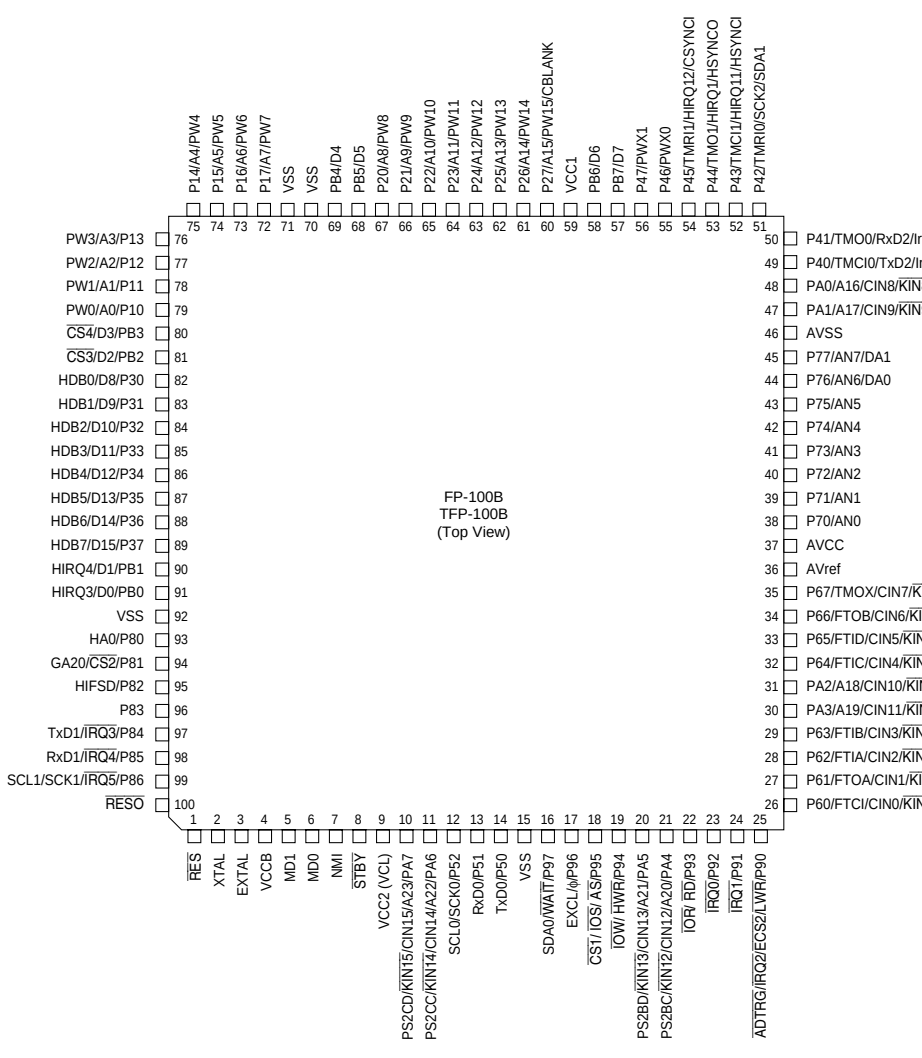


Figure 1.2 (a) Pin Arrangement of H8S/2148 Group (FP-100B, TFP-100B: To

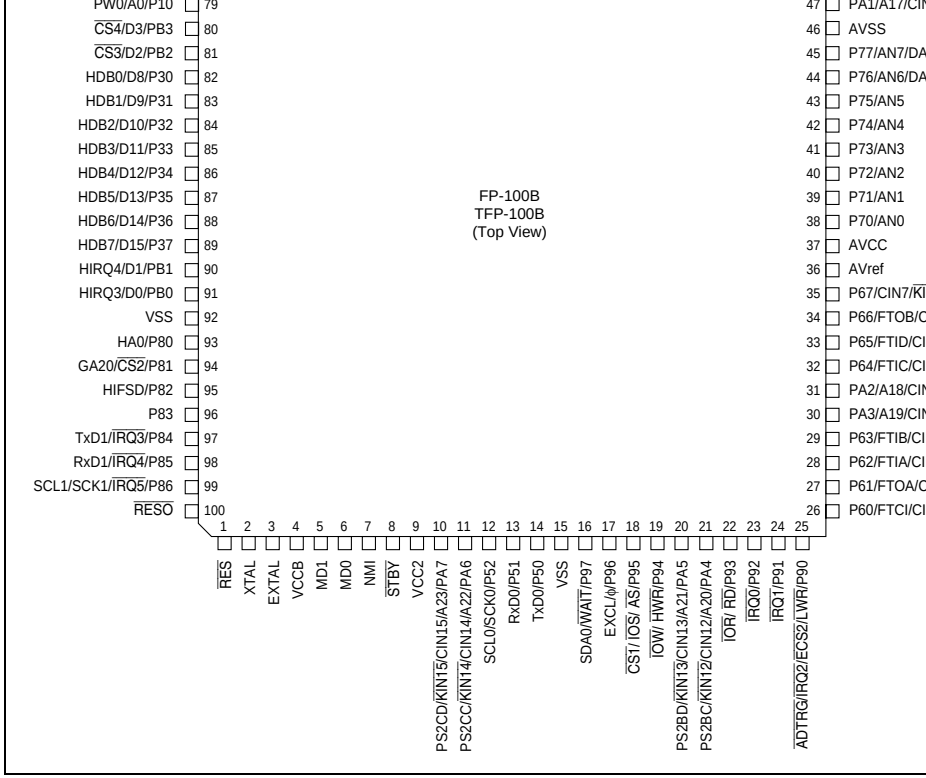


Figure 1.2 (b) Pin Arrangement of H8S/2147N (FP-100B, TFP-100B: Top

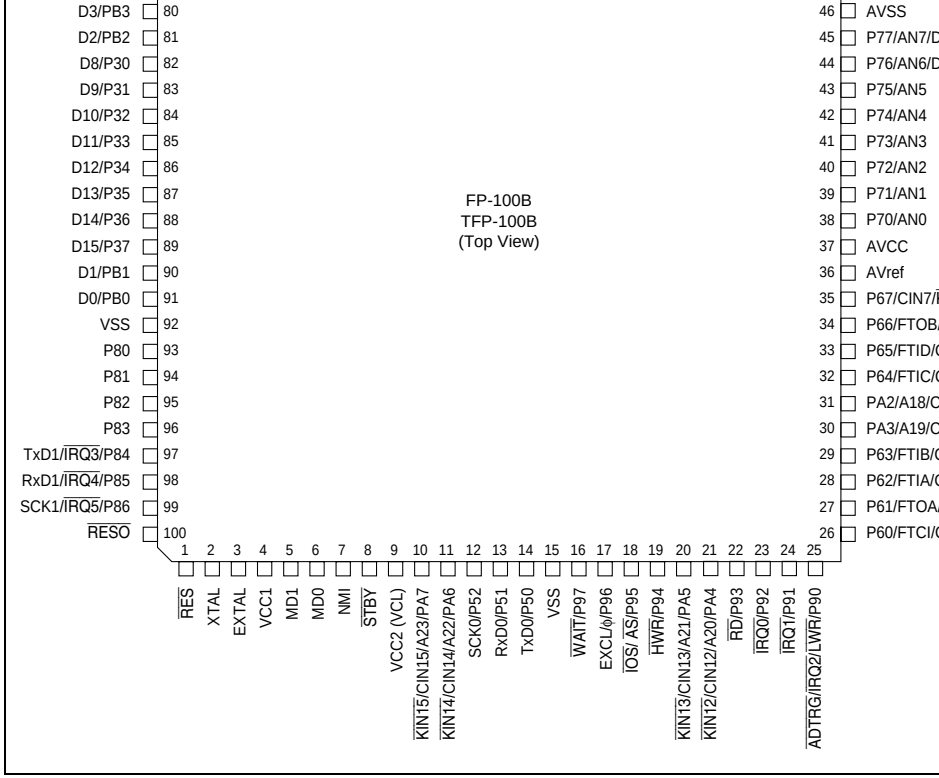


Figure 1.2 (c) Pin Arrangement of H8S/2144 Group (FP-100B, TFP-100B: To

FP-100B TFP-100B	Mode 1	Mode 2 (EXPE = 1) Mode 3 (EXPE = 1)	Mode 2 (EXPE = 0) Mode 3 (EXPE = 0)	Flas Wri
1	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$	$\overline{\text{RES}}$
2	XTAL	XTAL	XTAL	XTA
3	EXTAL	EXTAL	EXTAL	EXT
4	VCCB	VCCB	VCCB	VCC
5	MD1	MD1	MD1	VSS
6	MD0	MD0	MD0	VSS
7	NMI	NMI	NMI	FA9
8	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	VCC
9	VCC2 (VCL)	VCC2 (VCL)	VCC2 (VCL)	VCC
10	PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	A23/PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	NC
11	PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	A22/PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	NC
12	P52/SCK0/SCL0	P52/SCK0/SCL0	P52/SCK0/SCL0	NC
13	P51/RxD0	P51/RxD0	P51/RxD0	FA1
14	P50/TxD0	P50/TxD0	P50/TxD0	NC
15	VSS	VSS	VSS	VSS
16	P97/$\overline{\text{WAIT}}$/SDA0	P97/$\overline{\text{WAIT}}$/SDA0	P97/SDA0	VCC
17	P96/ϕ/EXCL	P96/ϕ/EXCL	P96/ϕ/EXCL	NC
18	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	P95/$\overline{\text{CS1}}$	FA1
19	$\overline{\text{HWR}}$	$\overline{\text{HWR}}$	P94/$\overline{\text{IOW}}$	FA1
20	PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	A21/PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	NC
21	PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	A20/PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	NC

25	LWR/P90/IRQ2/ ADTRG	LWR/P90/IRQ2/ ADTRG	P90/IRQ2/ADTRG/ ECS2	VCC
26	P60/FTCI/CIN0/ KIN0/TMIX/ HFBACKI	P60/FTCI/CIN0/ KIN0/TMIX/ HFBACKI	P60/FTCI/CIN0/ KIN0/TMIX/ HFBACKI	NC
27	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	P61/FTOA/CIN1/ KIN1/VSYNCO	NC
28	P62/FTIA/CIN2/ KIN2/TMIY/ VSYNCI	P62/FTIA/CIN2/ KIN2/TMIY/ VSYNCI	P62/FTIA/CIN2/ KIN2/TMIY/ VSYNCI	NC
29	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	P63/FTIB/CIN3/ KIN3/VFBACKI	NC
30	PA3/CIN11/ KIN11/PS2AD	A19/PA3/CIN11/ KIN11/PS2AD	PA3/CIN11/ KIN11/PS2AD	NC
31	PA2/CIN10/ KIN10/PS2AC	A18/PA2/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	VSS
36	AVref	AVref	AVref	VCC
37	AVCC	AVCC	AVCC	VCC
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	VSS
47	PA1/CIN9/$\overline{\text{KIN9}}$	A17/PA1/CIN9/ $\overline{\text{KIN9}}$	PA1/CIN9/$\overline{\text{KIN9}}$	NC
48	PA0/CIN8/$\overline{\text{KIN8}}$	A16/PA0/CIN8/$\overline{\text{KIN8}}$	PA0/CIN8/$\overline{\text{KIN8}}$	NC
49	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	NC
50	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	NC
51	P42/TMRI0/ SCK2/SDA1	P42/TMRI0/ SCK2/SDA1	P42/TMRI0/ SCK2/SDA1	NC
52	P43/TMCI1/ HSYNCI	P43/TMCI1/ HSYNCI	P43/TMCI1/HIRQ11/ HSYNCI	NC
53	P44/TMO1/ HSYNCO	P44/TMO1/ HSYNCO	P44/TMO1/HIRQ1/ HSYNCO	NC
54	P45/TMRI1/ CSYNCI	P45/TMRI1/ CSYNCI	P45/TMRI1/HIRQ12/ CSYNCI	NC
55	P46/PWX0	P46/PWX0	P46/PWX0	NC
56	P47/PWX1	P47/PWX1	P47/PWX1	NC
57	PB7/D7	PB7/D7	PB7	NC
58	PB6/D6	PB6/D6	PB6	NC
59	VCC1	VCC1	VCC1	VCC
60	A15	A15/P27/PW15/ CBLANK	P27/PW15/ CBLANK	$\overline{\text{CE}}$
61	A14	A14/P26/PW14	P26/PW14	FA1
62	A13	A13/P25/PW13	P25/PW13	FA1
63	A12	A12/P24/PW12	P24/PW12	FA1
64	A11	A11/P23/PW11	P23/PW11	FA1

68	PB5/D5	PB5/D5	PB5	NC
69	PB4/D4	PB4/D4	PB4	NC
70	VSS	VSS	VSS	VSS
71	VSS	VSS	VSS	VSS
72	A7	A7/P17/PW7	P17/PW7	FA7
73	A6	A6/P16/PW6	P16/PW6	FA6
74	A5	A5/P15/PW5	P15/PW5	FA5
75	A4	A4/P14/PW4	P14/PW4	FA4
76	A3	A3/P13/PW3	P13/PW3	FA3
77	A2	A2/P12/PW2	P12/PW2	FA2
78	A1	A1/P11/PW1	P11/PW1	FA1
79	A0	A0/P10/PW0	P10/PW0	FA0
80	PB3/D3	PB3/D3	PB3/$\overline{CS4}$	NC
81	PB2/D2	PB2/D2	PB2/$\overline{CS3}$	NC
82	D8	D8	P30/HDB0	FO0
83	D9	D9	P31/HDB1	FO1
84	D10	D10	P32/HDB2	FO2
85	D11	D11	P33/HDB3	FO3
86	D12	D12	P34/HDB4	FO4
87	D13	D13	P35/HDB5	FO5
88	D14	D14	P36/HDB6	FO6
89	D15	D15	P37/HDB7	FO7
90	PB1/D1	PB1/D1	PB1/HIRQ4	NC
91	PB0/D0	PB0/D0	PB0/HIRQ3	NC
92	VSS	VSS	VSS	VSS
93	P80	P80	P80/HA0	NC

97	P84/IRQ3/TxD1	P84/IRQ3/TxD1	P84/IRQ3/TxD1	NC
98	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	NC
99	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	NC
100	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	NC

3	EXTAL	EXTAL	EXTAL	EXTA
4	VCCB	VCCB	VCCB	VCC
5	MD1	MD1	MD1	VSS
6	MD0	MD0	MD0	VSS
7	NMI	NMI	NMI	FA9
8	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	VCC
9	VCC2	VCC2	VCC2	VCC
10	PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	A23/PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	PA7/CIN15/ $\overline{\text{KIN15}}$/PS2CD	NC
11	PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	A22/PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	PA6/CIN14/ $\overline{\text{KIN14}}$/PS2CC	NC
12	P52/SCK0/SCL0	P52/SCK0/SCL0	P52/SCK0/SCL0	NC
13	P51/RxD0	P51/RxD0	P51/RxD0	FA17
14	P50/TxD0	P50/TxD0	P50/TxD0	NC
15	VSS	VSS	VSS	VSS
16	P97/$\overline{\text{WAIT}}$/SDA0	P97/$\overline{\text{WAIT}}$/SDA0	P97/SDA0	VCC
17	ϕ/P96/EXCL	ϕ/P96/EXCL	P96/ϕ/EXCL	NC
18	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	P95/$\overline{\text{CS1}}$	FA16
19	$\overline{\text{HWR}}$	$\overline{\text{HWR}}$	P94/$\overline{\text{IOW}}$	FA15
20	PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	A21/PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	PA5/CIN13/ $\overline{\text{KIN13}}$/PS2BD	NC
21	PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	A20/PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	PA4/CIN12/ $\overline{\text{KIN12}}$/PS2BC	NC
22	$\overline{\text{RD}}$	$\overline{\text{RD}}$	P93/$\overline{\text{IOR}}$	$\overline{\text{WE}}$
23	P92/$\overline{\text{IRQ0}}$	P92/$\overline{\text{IRQ0}}$	P92/$\overline{\text{IRQ0}}$	VSS
24	P91/$\overline{\text{IRQ1}}$	P91/$\overline{\text{IRQ1}}$	P91/$\overline{\text{IRQ1}}$	VCC

27	P61/FTOA/CIN1/ $\overline{\text{KIN1}}$	P61/FTOA/CIN1/ $\overline{\text{KIN1}}$	P61/FTOA/CIN1/ $\overline{\text{KIN1}}$	NC
28	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	NC
29	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	NC
30	PA3/CIN11/ $\overline{\text{KIN11}}$/PS2AD	A19/PA3/CIN11/ $\overline{\text{KIN11}}$/PS2AD	PA3/CIN11/ $\overline{\text{KIN11}}$/PS2AD	NC
31	PA2/CIN10/ $\overline{\text{KIN10}}$/PS2AC	A18/PA2/CIN10/ $\overline{\text{KIN10}}$/PS2AC	PA2/CIN10/ $\overline{\text{KIN10}}$/PS2AC	NC
32	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	NC
33	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	NC
34	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	NC
35	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	VSS
36	AVref	AVref	AVref	VCC
37	AVCC	AVCC	AVCC	VCC
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
42	P74/AN4	P74/AN4	P74/AN4	NC
43	P75/AN5	P75/AN5	P75/AN5	NC
44	P76/AN6/DA0	P76/AN6/DA0	P76/AN6/DA0	NC
45	P77/AN7/DA1	P77/AN7/DA1	P77/AN7/DA1	NC

49	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	NC
50	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	NC
51	P42/TMRI0/ SCK2/SDA1	P42/TMRI0/ SCK2/SDA1	P42/TMRI0/ SCK2/SDA1	NC
52	P43/TMCI1	P43/TMCI1	P43/TMCI1/HIRQ11	NC
53	P44/TMO1	P44/TMO1	P44/TMO1/HIRQ1	NC
54	P45/TMRI1	P45/TMRI1	P45/TMRI1/HIRQ12	NC
55	P46/PWX0	P46/PWX0	P46/PWX0	NC
56	P47/PWX1	P47/PWX1	P47/PWX1	NC
57	PB7/D7	PB7/D7	PB7	NC
58	PB6/D6	PB6/D6	PB6	NC
59	VCC1	VCC1	VCC1	VCC
60	A15	A15/P27/PW15	P27/PW15	$\overline{CE}$
61	A14	A14/P26/PW14	P26/PW14	FA14
62	A13	A13/P25/PW13	P25/PW13	FA13
63	A12	A12/P24/PW12	P24/PW12	FA12
64	A11	A11/P23/PW11	P23/PW11	FA11
65	A10	A10/P22/PW10	P22/PW10	FA10
66	A9	A9/P21/PW9	P21/PW9	$\overline{OE}$
67	A8	A8/P20/PW8	P20/PW8	FA8
68	PB5/D5	PB5/D5	PB5	NC
69	PB4/D4	PB4/D4	PB4	NC
70	VSS	VSS	VSS	VSS
71	VSS	VSS	VSS	VSS

75	A4	A4/P14/PW4	P14/PW4	FA4
76	A3	A3/P13/PW3	P13/PW3	FA3
77	A2	A2/P12/PW2	P12/PW2	FA2
78	A1	A1/P11/PW1	P11/PW1	FA1
79	A0	A0/P10/PW0	P10/PW0	FA0
80	PB3/D3	PB3/D3	PB3/$\overline{\text{CS4}}$	NC
81	PB2/D2	PB2/D2	PB2/$\overline{\text{CS3}}$	NC
82	D8	D8	P30/HDB0	FO0
83	D9	D9	P31/HDB1	FO1
84	D10	D10	P32/HDB2	FO2
85	D11	D11	P33/HDB3	FO3
86	D12	D12	P34/HDB4	FO4
87	D13	D13	P35/HDB5	FO5
88	D14	D14	P36/HDB6	FO6
89	D15	D15	P37/HDB7	FO7
90	PB1/D1	PB1/D1	PB1/HIRQ4	NC
91	PB0/D0	PB0/D0	PB0/HIRQ3	NC
92	VSS	VSS	VSS	VSS
93	P80	P80	P80/HA0	NC
94	P81	P81	P81/$\overline{\text{CS2}}$/GA20	NC
95	P82	P82	P82/HIFSD	NC
96	P83	P83	P83	NC
97	P84/$\overline{\text{IRQ3}}$/TxD1	P84/$\overline{\text{IRQ3}}$/TxD1	P84/$\overline{\text{IRQ3}}$/TxD1	NC
98	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	NC
99	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	P86/$\overline{\text{IRQ5}}$/SCK1/ SCL1	NC
100	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	NC

3	EXTAL	EXTAL	EXTAL	EXTA
4	VCC1	VCC1	VCC1	VCC
5	MD1	MD1	MD1	VSS
6	MD0	MD0	MD0	VSS
7	NMI	NMI	NMI	FA9
8	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	$\overline{\text{STBY}}$	VCC
9	VCC2 (VCL)	VCC2 (VCL)	VCC2 (VCL)	VCC
10	PA7/CIN15/ $\overline{\text{KIN15}}$	A23/PA7/CIN15/ $\overline{\text{KIN15}}$	PA7/CIN15/ $\overline{\text{KIN15}}$	NC
11	PA6/CIN14/ $\overline{\text{KIN14}}$	A22/PA6/CIN14/ $\overline{\text{KIN14}}$	PA6/CIN14/ $\overline{\text{KIN14}}$	NC
12	P52/SCK0	P52/SCK0	P52/SCK0	NC
13	P51/RxD0	P51/RxD0	P51/RxD0	FA17
14	P50/TxD0	P50/TxD0	P50/TxD0	NC
15	VSS	VSS	VSS	VSS
16	P97/$\overline{\text{WAIT}}$	P97/$\overline{\text{WAIT}}$	P97	VCC
17	ϕ/P96/EXCL	ϕ/P96/EXCL	P96/ϕ/EXCL	NC
18	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	P95	FA16
19	$\overline{\text{HWR}}$	$\overline{\text{HWR}}$	P94	FA15
20	PA5/CIN13/ $\overline{\text{KIN13}}$	A21/PA5/CIN13/ $\overline{\text{KIN13}}$	PA5/CIN13/ $\overline{\text{KIN13}}$	NC
21	PA4/CIN12/ $\overline{\text{KIN12}}$	A20/PA4/CIN12/ $\overline{\text{KIN12}}$	PA4/CIN12/$\overline{\text{KIN12}}$	NC
22	$\overline{\text{RD}}$	$\overline{\text{RD}}$	P93	$\overline{\text{WE}}$
23	P92/$\overline{\text{IRQ0}}$	P92/$\overline{\text{IRQ0}}$	P92/$\overline{\text{IRQ0}}$	VSS
24	P91/$\overline{\text{IRQ1}}$	P91/$\overline{\text{IRQ1}}$	P91/$\overline{\text{IRQ1}}$	VCC

27	P61/FTOA/ CIN1/$\overline{\text{KIN1}}$	P61/FTOA/CIN1/ $\overline{\text{KIN1}}$	P61/FTOA/CIN1/ $\overline{\text{KIN1}}$	NC
28	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	P62/FTIA/CIN2/ $\overline{\text{KIN2}}$/TMIY	NC
29	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	P63/FTIB/CIN3/ $\overline{\text{KIN3}}$	NC
30	PA3/CIN11/ $\overline{\text{KIN11}}$	A19/PA3/CIN11/ $\overline{\text{KIN11}}$	PA3/CIN11/ $\overline{\text{KIN11}}$	NC
31	PA2/CIN10/ $\overline{\text{KIN10}}$	A18/PA2/CIN10/ $\overline{\text{KIN10}}$	PA2/CIN10/ $\overline{\text{KIN10}}$	NC
32	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	P64/FTIC/CIN4/ $\overline{\text{KIN4}}$	NC
33	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	P65/FTID/CIN5/ $\overline{\text{KIN5}}$	NC
34	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	P66/FTOB/CIN6/ $\overline{\text{KIN6}}$/$\overline{\text{IRQ6}}$	NC
35	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	P67/CIN7/$\overline{\text{KIN7}}$/ $\overline{\text{IRQ7}}$	VSS
36	AVref	AVref	AVref	VCC
37	AVCC	AVCC	AVCC	VCC
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
42	P74/AN4	P74/AN4	P74/AN4	NC
43	P75/AN5	P75/AN5	P75/AN5	NC
44	P76/AN6/DA0	P76/AN6/DA0	P76/AN6/DA0	NC
45	P77/AN7/DA1	P77/AN7/DA1	P77/AN7/DA1	NC

49	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	P40/TMCI0/ TxD2/IrTxD	NC
50	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	P41/TMO0/ RxD2/IrRxD	NC
51	P42/TMRI0/ SCK2	P42/TMRI0/ SCK2	P42/TMRI0/ SCK2	NC
52	P43/TMCI1	P43/TMCI1	P43/TMCI1	NC
53	P44/TMO1	P44/TMO1	P44/TMO1	NC
54	P45/TMRI1	P45/TMRI1	P45/TMRI1	NC
55	P46/PWX0	P46/PWX0	P46/PWX0	NC
56	P47/PWX1	P47/PWX1	P47/PWX1	NC
57	PB7/D7	PB7/D7	PB7	NC
58	PB6/D6	PB6/D6	PB6	NC
59	VCC1	VCC1	VCC1	VCC
60	A15	A15/P27	P27	$\overline{CE}$
61	A14	A14/P26	P26	FA14
62	A13	A13/P25	P25	FA13
63	A12	A12/P24	P24	FA12
64	A11	A11/P23	P23	FA11
65	A10	A10/P22	P22	FA10
66	A9	A9/P21	P21	$\overline{OE}$
67	A8	A8/P20	P20	FA8
68	PB5/D5	PB5/D5	PB5	NC
69	PB4/D4	PB4/D4	PB4	NC
70	VSS	VSS	VSS	VSS
71	VSS	VSS	VSS	VSS

75	A4	A4/P14	P14	FA4
76	A3	A3/P13	P13	FA3
77	A2	A2/P12	P12	FA2
78	A1	A1/P11	P11	FA1
79	A0	A0/P10	P10	FA0
80	PB3/D3	PB3/D3	PB3	NC
81	PB2/D2	PB2/D2	PB2	NC
82	D8	D8	P30	FO0
83	D9	D9	P31	FO1
84	D10	D10	P32	FO2
85	D11	D11	P33	FO3
86	D12	D12	P34	FO4
87	D13	D13	P35	FO5
88	D14	D14	P36	FO6
89	D15	D15	P37	FO7
90	PB1/D1	PB1/D1	PB1	NC
91	PB0/D0	PB0/D0	PB0	NC
92	VSS	VSS	VSS	VSS
93	P80	P80	P80	NC
94	P81	P81	P81	NC
95	P82	P82	P82	NC
96	P83	P83	P83	NC
97	P84/$\overline{\text{IRQ3}}$/TxD1	P84/$\overline{\text{IRQ3}}$/TxD1	P84/$\overline{\text{IRQ3}}$/TxD1	NC
98	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	P85/$\overline{\text{IRQ4}}$/RxD1	NC
99	P86/$\overline{\text{IRQ5}}$/SCK1	P86/$\overline{\text{IRQ5}}$/SCK1	P86/$\overline{\text{IRQ5}}$/SCK1	NC
100	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	$\overline{\text{RESO}}$	NC

Power supply	VCC1 VCC2	4 [H8S/2144 Group only], 59 9*	Input	**Power supply:** For connection to the po All VCC1 and VCC2* pins should be co the system power supply.
	VCL	9*	Input	**Internal step-down voltage pin:** A pow pin for the product, applicable to produc have an internal step-down voltage. In th 4-V versions, connect external capacitor stabilize the internal step-down voltage this pin and the VSS pin. Do not connec In the 3-V version, connect this pin and pin to the power supply for the system. See section 26, Electrical Characteristic
	VCCB	4 [H8S/2148 Group and H8S/2147N only]	Input	**Input/output buffer power supply:** Po pin for the port A input/output buffer.
	VSS	15, 70 71, 92	Input	**Ground:** All VSS pins should be connec system power supply (0 V).
Clock	XTAL	2	Input	Connected to a crystal oscillator. See se Clock Pulse Generator, for typical conne diagrams for a crystal oscillator and exte input.
	EXTAL	3	Input	Connected to a crystal oscillator. The EX can also input an external clock. See se Clock Pulse Generator, for typical conne diagrams for a crystal oscillator and exte input.
	ϕ	17	Output	**System clock:** Supplies the system clo external devices.
	EXCL	17	Input	**External subclock input:** Input a 32.76 external subclock.

MD1	MD0	Mode	Description
0	1	Mode 1	Normal Expanded mo on-chip ROM
1	0	Mode 2	Advanced Expanded mo on-chip ROM single-chip m
1	1	Mode 3	Normal Expanded mo on-chip ROM single-chip m

System control	$\overline{\text{RES}}$	1	Input	**Reset input:** When this pin is driven lo reset.
	$\overline{\text{RESO}}$	100	Output	**Reset output:** Outputs reset signal to device.
	$\overline{\text{STBY}}$	8	Input	**Standby:** When this pin is driven low, made to hardware standby mode.
Address bus	A23 to A16	10, 11, 20, 21, 30, 31, 47, 48	Output	**Address bus (advanced):** Outputs ad 16-Mbyte space is used.
	A15 to A0	60 to 67, 72 to 79	Output	**Address bus:** These pins output an ad
Data bus	D15 to D8	89 to 82	Input/ output	**Data bus (upper):** Bidirectional data b Used for 8-bit data and upper byte of 1
	D7 to D0	57, 58, 68, 69, 80, 81, 90, 91	Input/ output	**Data bus (lower):** Bidirectional data bu Used for lower byte of 16-bit data.

	HWR	19	Output	**High write:** When this pin is low, it indic the external address space is being writ upper half of the data bus is valid.
	$\overline{\text{LWR}}$	25	Output	**Low write:** When this pin is low, it indica external address space is being written lower half of the data bus is valid.
	$\overline{\text{AS}}$/$\overline{\text{IOS}}$	18	Output	**Address strobe:** When this pin is low, i that address output on the address bus
Interrupt signals	NMI	7	Input	**Nonmaskable interrupt:** Requests a n interrupt.
	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ7}}$	23 to 25, 97 to 99, 34, 35	Input	**Interrupt request 0 to 7:** These pins re maskable interrupt.
16-bit free-running timer (FRT)	FTCI	26	Input	**FRT counter clock input:** Input pin for clock signal for the free-running counter
	FTOA	27	Output	**FRT output compare A output:** The ou compare A output pin.
	FTOB	34	Output	**FRT output compare B output:** The ou compare B output pin.
	FTIA	28	Input	**FRT input capture A input:** The input c input pin.
	FTIB	29	Input	**FRT input capture B input:** The input c input pin.
	FTIC	32	Input	**FRT input capture C input:** The input c input pin.
	FTID	33	Input	**FRT input capture D input:** The input c input pin.

				counters.
	TMRI0 TMRI1	51 54	Input	**Counter external reset input:** TMR0 a counter reset input pins.
	TMIX TMIY	26 28	Input	**Counter external clock input and res** Dual function as TMRX and TMRY cou input pin and reset input pin.
PWM timer (PWM)	PW15 to PW0	60 to 67, 72 to 79	Output	**PWM timer output:** PWM timer pulse
14-bit PWM timer (PWMX)	PWX0 PWX1	55 56	Output	**PWMX timer output:** PWM D/A pulse
Serial communication interface (SCI0, SCI1, SCI2)	TxD0 TxD1 TxD2	14 97 49	Output	**Transmit data:** Data output pins.
	RxD0 RxD1 RxD2	13 98 50	Input	**Receive data:** Data input pins.
	SCK0 SCK1 SCK2	12 99 51	Input/ output	**Serial clock:** Clock input/output pins. The SCK0 output type is NMOS push-p H8S/2148 Group and H8S/2147N, and output in the H8S/2144 Group.
SCI with IrDA (SCI2)	IrTxD IrRxD	49 50	Output Input	**IrDA transmit data/receive data:** Inpu pins for data encoded for IrDA use.
Keyboard Buffer controller (PS2)	PS2AC PS2BC PS2CC	31 21 11	Input/ output	**PS2 clock:** Keyboard buffer controller synchronization clock input/output pins
	PS2AD PS2BD PS2CD	30 20 10	Input/ output	**PS2 data:** Keyboard buffer controller d input/output pins.

	$\overline{CS0}$, $\overline{CS4}$			
	$\overline{IOR}$	22	Input	**I/O read:** Input pin that enables reading host interface.
	$\overline{IOW}$	19	Input	**I/O write:** Input pin that enables writing interface.
	HA0	93	Input	**Command/data:** Input pin that indicates an access is a data access or command
	GA20	94	Output	**GATE A20:** A20 gate control signal outp
	HIRQ11 HIRQ1 HIRQ12 HIRQ3 HIRQ4	52 53 54 91 90	Output	**Host interrupt 11, 1, 12, 3, and 4:** Outp interrupt requests to the host.
	HIFSD	95	Input	**Host interface shutdown:** Control inpu to place host interface input/output pins impedance/cutoff state.
Keyboard control	$\overline{KIN0}$ to $\overline{KIN15}$	26 to 29, 32 to 35, 48, 47, 31, 30, 21, 20, 11, 10	Input	**Keyboard input:** Matrix keyboard input Normally, P10 to P17 and P20 to P27 a key-scan outputs. This enables a maxim output × 16-input, 256-key matrix to be
A/D converter (ADC)	AN7 to AN0	45 to 38	Input	**Analog input:** A/D converter analog inp
	CIN0 to CIN15	26 to 29, 32 to 35, 48, 47, 31, 30, 21, 20, 11, 10	Input	**Expansion A/D inputs:** Expansion A/D can be connected to the A/D converter, they are also used as digital input/outpu precision will fall.
	$\overline{ADTRG}$	25	Input	**A/D conversion external trigger input** input of an external trigger to start A/D c

D/A converter				When the A/D and D/A converters are pin should be connected to the system supply (+5 V or +3 V).
	AVref	36	Input	**Analog reference voltage:** The refere supply pin for the A/D converter and D/ When the A/D and D/A converters are pin should be connected to the system supply (+5 V or +3 V).
	AVSS	46	Input	**Analog ground:** The ground pin for th converter and D/A converter. This pin s connected to the system power supply
Timer connection	VSYNCI, HSYNCI, CSYNCI, VFBACKI, HFBACKI	28 52 54 29 26	Input	**Timer connection input:** Timer conne synchronous signal input pins.
	VSYNCO, HSYNCO, CLAMPO, CBLANK	27 53 32 60	Output	**Timer connection output:** Timer conn synchronous signal output pins.
I^2C bus interface (IIC) (option)	SCL0 SCL1	12 99	Input/ output	**I^2C clock input/output (channels 0 a** clock I/O pins. These pins have a bus function. The SCL0 output form is NMOS open-
	SDA0 SDA1	16 51	Input/ output	**I^2C data input/output (channels 0 an** I/O pins. These pins have a bus drive f The SDA0 output form is NMOS open-

			register (P2DDR). These pins have built input pull-ups, and also have LED drive
P37 to P30	89 to 82	Input/ output	**Port 3:** Eight input/output pins. The data each pin can be selected in the port 3 d register (P3DDR). These pins have built input pull-ups, and also have LED drive
P47 to P40	56 to 49	Input/ output	**Port 4:** Eight input/output pins. The data each pin can be selected in the port 4 d register (P4DDR).
P52 to P50	12 to 14	Input/ output	**Port 5:** Three input/output pins. The dat of each pin can be selected in the port 5 direction register (P5DDR). P52 is an N pull output in the H8S/2148 Group and and is a CMOS output in the H8S/2144
P67 to P60	35 to 32 29 to 26	Input/ output	**Port 6:** Eight input/output pins. The data each pin can be selected in the port 6 d register (P6DDR). These pins have built input pull-ups.
P77 to P70	45 to 38	Input	**Port 7:** Eight input pins.
P86 to P80	99 to 93	Input/ output	**Port 8:** Seven input/output pins. The da of each pin can be selected in the port 8 direction register (P8DDR).
P97 to P90	16 to 19 22 to 25	Input/ output	**Port 9:** Eight input/output pins. The data each pin (except P96) can be selected i data direction register (P9DDR). P97 is push-pull output in the H8S/2148 Group H8S/2147N, and is a CMOS output in th Group.

	PB7 to PB0	69, 80, 81, 90, 91	Input/ output	**Port B:** Eight input/output pins. The di of each pin can be selected in the port direction register (PBDDR). These pins MOS input pull-ups.

Note: * In F-ZTAT and mask ROM versions of HD64F2148A, HD64F2147A, HD64
HD6432148S, HD6432148SW, HD6432147S, HD6432147SW, HD643214
HD6432143S pin NO.9 is VCL pin and is not VCC pin.

ideal for realtime control.

2.1.1 Features

The H8S/2000 CPU has the following features.

- Upward-compatible with H8/300 and H8/300H CPUs
  - Can execute H8/300 and H8/300H object programs
- General-register architecture
  - Sixteen 16-bit general registers (also usable as sixteen 8-bit registers or eight 3 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 (4 Gbytes architecturally)

— 32 ÷ 16-bit register-register divide: 1000 ns

- Two CPU operating modes
  - — Normal mode
  - — Advanced mode
- Power-down state
  - — Transition to power-down state by SLEEP instruction
  - — CPU clock speed selection

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

The differences between the H8S/2600 CPU and the H8S/2000 CPU are 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
  H8S/2600 CPU.
- Number of execution states

  The number of execution states of the MULXU and MULXS instructions differ as f

		Number of Execution Stat	
Instruction	**Mnemonic**	**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

There are also differences in the address space, EXR register functions, power-down st
depending on the product.

— Advanced mode supports a maximum 16-Mbyte address space.

- Enhanced addressing
  — The addressing modes have been enhanced to make effective use of the 16-Mb space.
- Enhanced instructions
  — Addressing modes of bit-manipulation instructions have been enhanced.
  — Signed multiply and divide instructions have been added.
  — Two-bit shift 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 execute twice as fast.

2.1.4 Differences from H8/300H CPU

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

- 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 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 execute twice as fast.

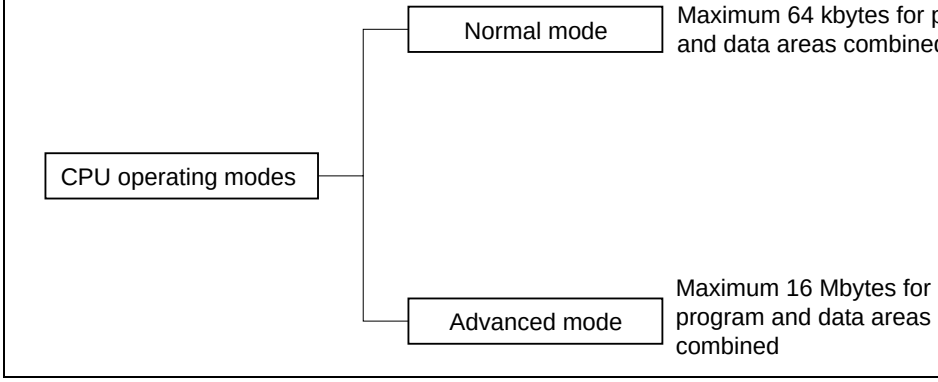


Figure 2.1 CPU Operating Modes

(1) Normal Mode

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

Address Space: A maximum address space of 64 kbytes can be accessed.

Extended Registers (En): The extended registers (E0 to E7) can be used as 16-bit regi
the upper 16-bit segments of 32-bit registers. When En is used as a 16-bit register it can
any value, even when the corresponding general register (Rn) is used as an address regi
general register is referenced in the register indirect addressing mode with pre-decreme
or post-increment (@Rn+) and a carry or borrow occurs, however, the value in the corr
extended register (En) will be affected.

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

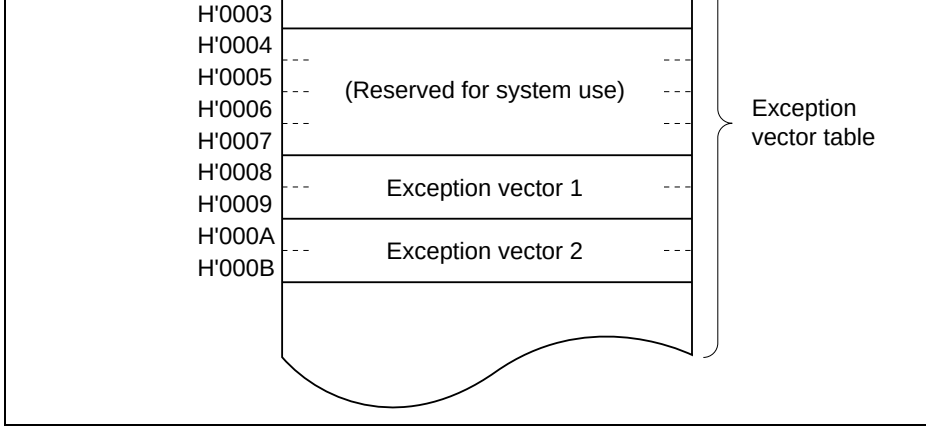


Figure 2.2 Exception Vector Table (Normal Mode)

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

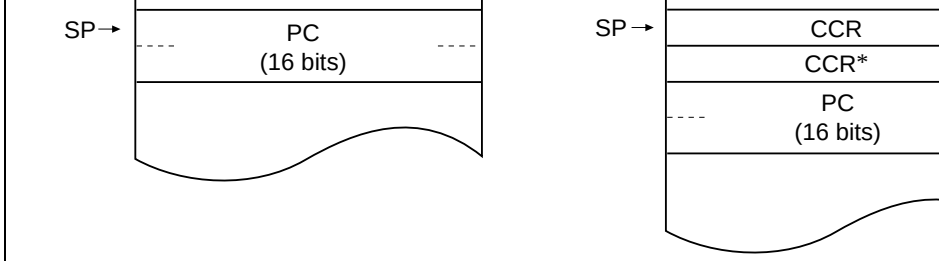


Figure 2.3 Stack Structure in Normal Mode

(2) Advanced Mode

Address Space: Linear access is provided to a 16-Mbyte maximum address space (arch a maximum 16-Mbyte program area and a maximum 4-Gbyte data area, with a maximu Gbytes for program and data areas combined).

Extended Registers (En): The extended registers (E0 to E7) can be used as 16-bit regi the upper 16-bit segments of 32-bit registers or address registers.

Instruction Set: All instructions and addressing modes can be used.

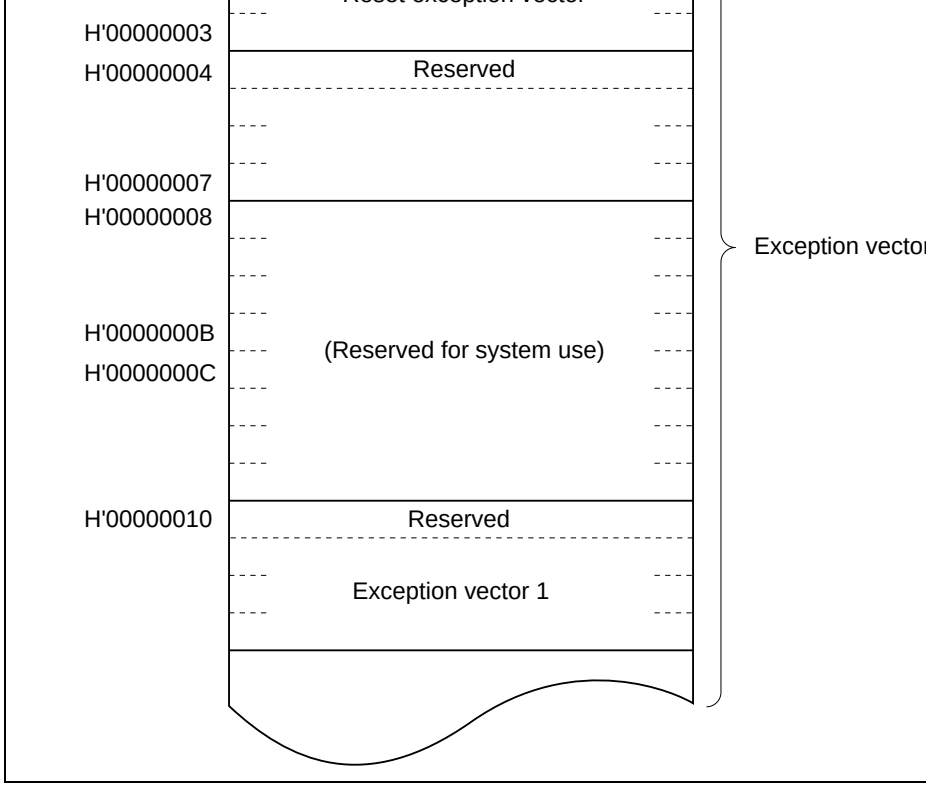


Figure 2.4 Exception Vector Table (Advanced Mode)

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instru
an 8-bit absolute address included in the instruction code to specify a memory operand
contains a branch address. In advanced mode the operand is a 32-bit longword operan
a 32-bit branch address. The upper 8 bits of these 32 bits are a reserved area that is reg
H'00. Branch addresses can be stored in the area from H'00000000 to H'000000FF. No
first part of this range is also the exception vector table.

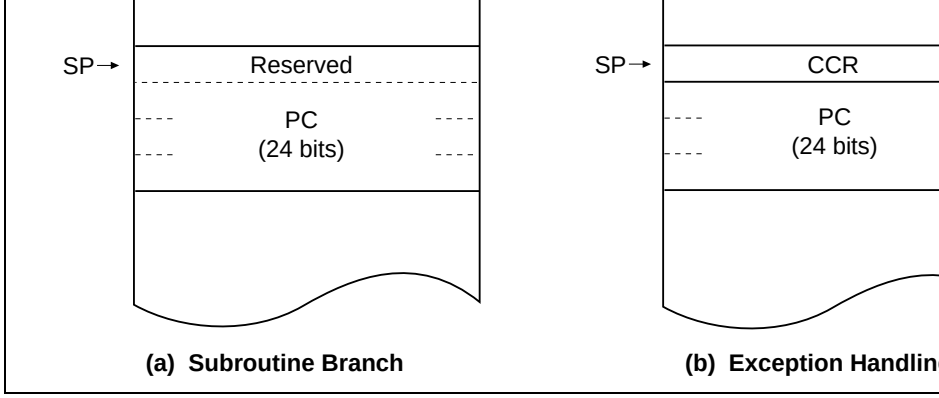


Figure 2.5 Stack Structure in Advanced Mode

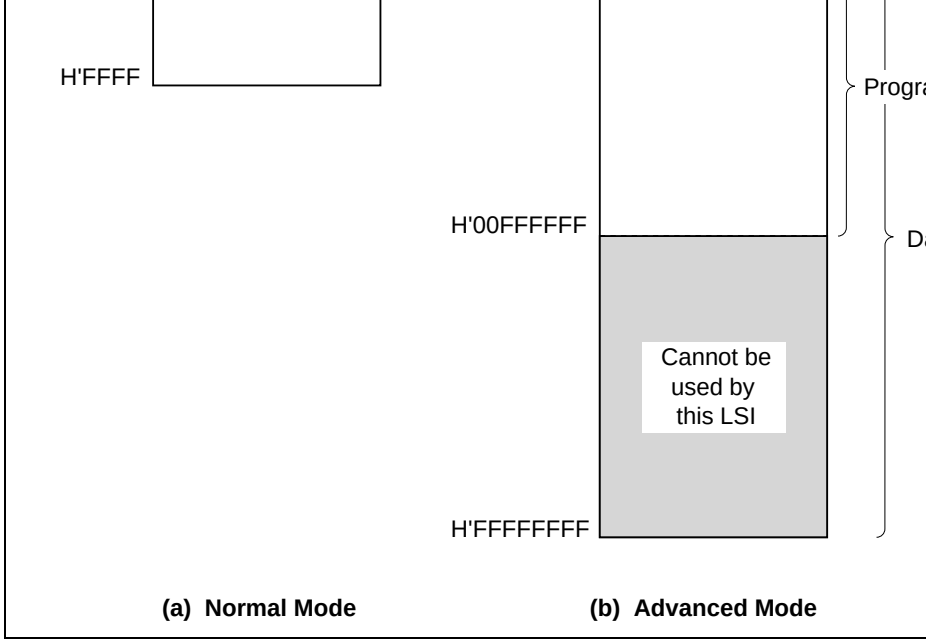
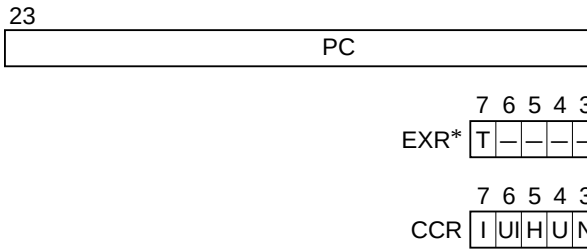


Figure 2.6 Memory Map

General Registers (Rn) and Extended Registers (En)

	15 0	7 0	7
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 (CR)



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 2.7 CPU Registers

registers. The E registers (E0 to E7) are also referred to as extended registers.

The R registers divide into 8-bit general registers designated by the letters RH (R0H to
RL (R0L to R7L). These registers are functionally equivalent, providing a maximum o
bit registers.

Figure 2.8 illustrates the usage of the general registers. The usage of each register can
independently.

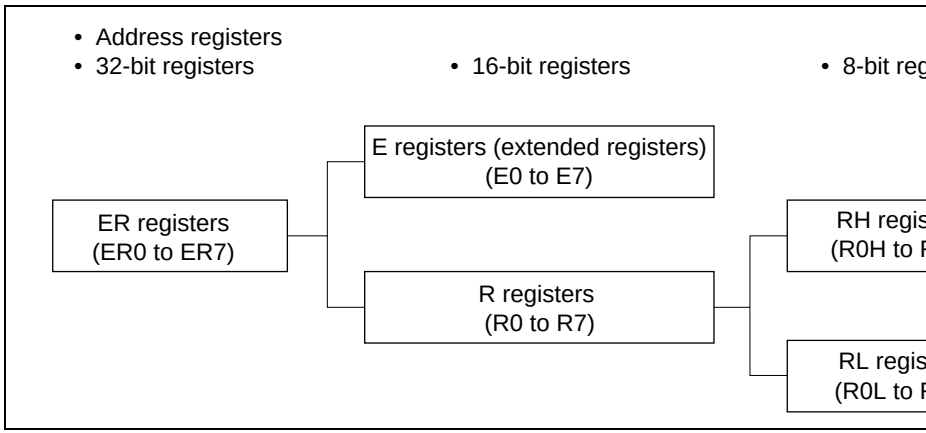


Figure 2.8 Usage of General Registers

General register ER7 has the function of stack pointer (SP) in addition to its general-r
function, and is used implicitly in exception handling and subroutine calls. Figure 2.9
stack.

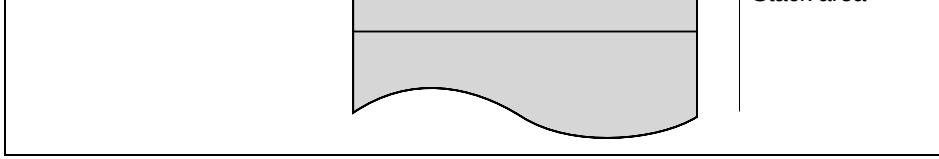


Figure 2.9 Stack

2.4.3 Control Registers

The control registers are the 24-bit program counter (PC), 8-bit extended control registe and 8-bit condition-code register (CCR).

(1) Program Counter (PC)

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

(2) Extended Control Register (EXR)

An 8-bit register. In this LSI, this register does not affect operation.

Bit 7—Trace Bit (T): This bit is reserved. In this LSI, this bit does not affect operation

Bits 6 to 3—Reserved: These bits are reserved. They are always read as 1.

Bits 2 to 0—Interrupt Mask Bits (I2 to I0): These bits are reserved. In this LSI, these affect operation.

Bit 6—User Bit or Interrupt Mask Bit (UI): Can be written and read by software us
LDC, STC, ANDC, ORC, and XORC instructions. This bit can also be used as an inte
bit. For details, refer to section 5, Interrupt Controller.

Bit 5—Half-Carry Flag (H): When the ADD.B, ADDX.B, SUB.B, SUBX.B, CMP.
instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and cle
otherwise. When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, t
set to 1 if there is a carry or borrow at bit 11, and cleared to 0 otherwise. When the AD
SUB.L, CMP.L, or NEG.L instruction is executed, the H flag is set to 1 if there is a ca
borrow at bit 27, and cleared to 0 otherwise.

Bit 4—User Bit (U): Can be written and read by software using the LDC, STC, AND
XORC instructions.

Bit 3—Negative Flag (N): Stores the value of the most significant bit (sign bit) of dat

Bit 2—Zero Flag (Z): Set to 1 to indicate zero data, and cleared to 0 to indicate non-z

Bit 1—Overflow Flag (V): Set to 1 when an arithmetic overflow occurs, and cleared
otherwise.

Bit 0—Carry Flag (C): Set to 1 when a carry occurs, and cleared to 0 otherwise. Use
- Add instructions, to indicate a carry
- Subtract instructions, to indicate a borrow
- Shift and rotate instructions, to store the carry

The carry flag is also used as a bit accumulator by bit-manipulation instructions.

Some instructions leave some or all of the flag bits unchanged. For the action of each
on the flag bits, refer to appendix A.1, Instruction.

Operations can be performed on the CCR bits by the LDC, STC, ANDC, ORC, and X
instructions. The N, Z, V, and C flags are used as branching conditions for conditiona
(Bcc) instructions.

Figure 2.10 shows the data formats in general registers.

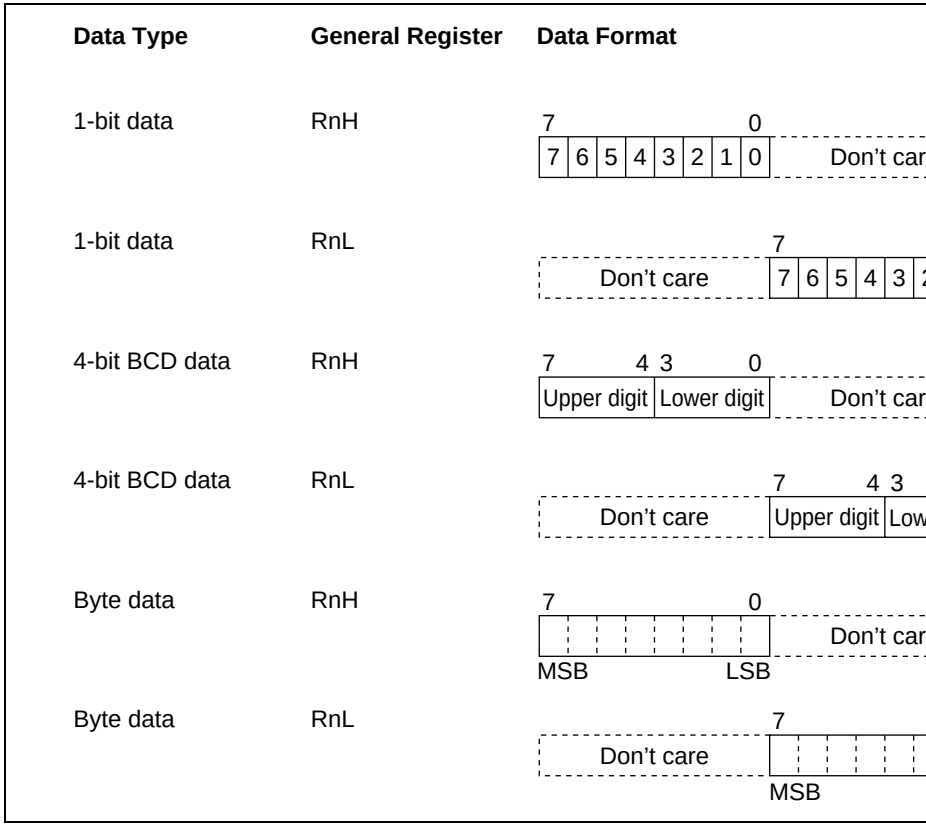
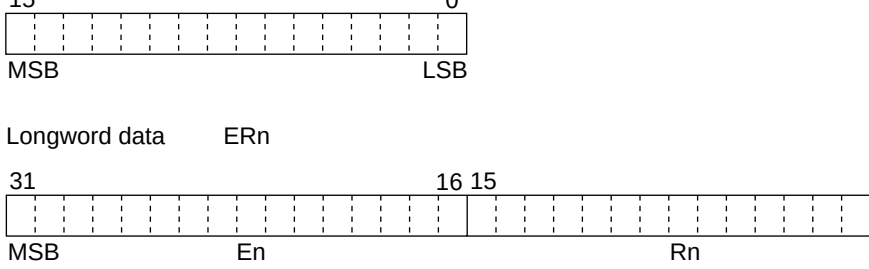


Figure 2.10 General Register Data Formats



Legend:

ERn: General register ER
En: General register E
Rn: General register R
RnH: General register RH
RnL: General register RL
MSB: Most significant bit
LSB: Least significant bit

Figure 2.10 General Register Data Formats (cont)

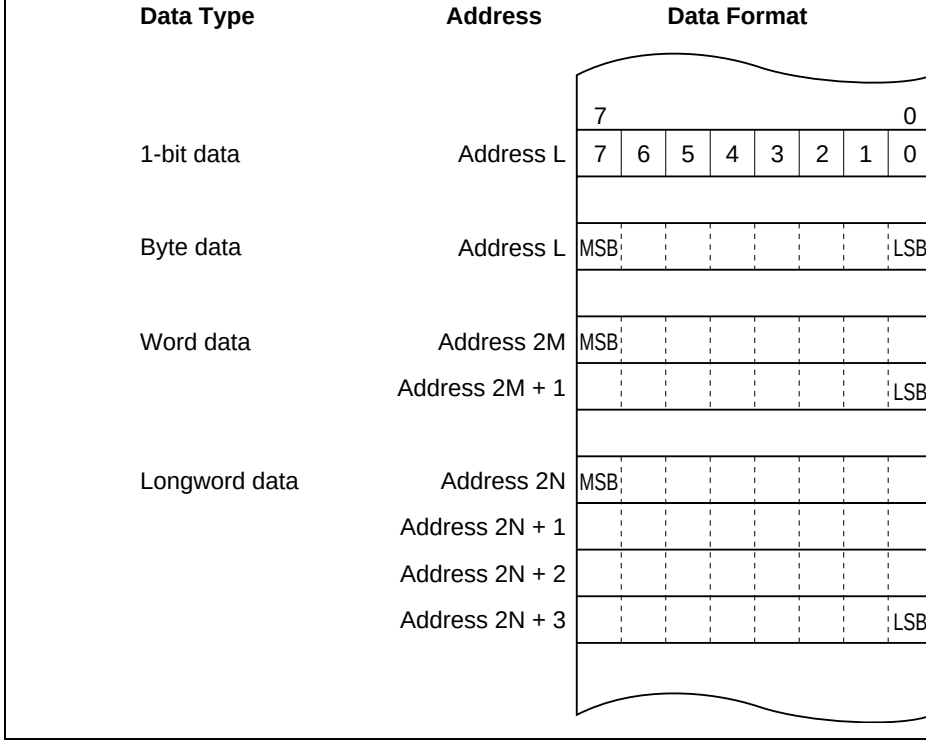


Figure 2.11 Memory Data Formats

When ER7 (SP) is used as an address register to access the stack, the operand size sho
size or longword size.

Function	Instructions	Size
Data transfer	MOV	BW
	POP*1, PUSH*1	WL
	LDM*5, STM*5	L
	MOVFPE*3, MOVTPE*3	B
Arithmetic operations	ADD, SUB, CMP, NEG	BW
	ADDX, SUBX, DAA, DAS	B
	INC, DEC	BW
	ADDS, SUBS	L
	MULXU, DIVXU, MULXS, DIVXS	BW
	EXTU, EXTS	WL
	TAS*4	B
Logic operations	AND, OR, XOR, NOT	BW
Shift	SHAL, SHAR, SHLL, SHLR, ROTL, ROTR, ROTXL, ROTXR	BW
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	—

To

Legend: B: Byte
W: Word
L: Longword

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. @-SP.
2. Bcc is the general name for conditional branch instructions.
3. Cannot be used in this LSI.
4. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS in
5. Only registers ER0 to ER6 should be used when using the STM/LDM instruc

Function	Instruction	#xx	Rn	@ERn	@(d:16,ERn)	@(d:32, ERn	@–ERn/@ER	@aa:8	@aa:16	@aa:24	@aa:32	@(d:8, PC)	@(d:16, PC)
Data transfer	MOV	BWL	BWL	BWL	BWL	BWL	BWL	B	BWL	—	BWL	—	–
	POP, PUSH	—	—	—	—	—	—	—	—	—	—	—	–
	LDM*3, STM*3	—	—	—	—	—	—	—	—	—	—	—	–
	MOVFPE*1, MOVTPE*1	—	—	—	—	—	—	—	B	—	—	—	–
Arithmetic operations	ADD, CMP	BWL	BWL	—	—	—	—	—	—	—	—	—	–
	SUB	WL	BWL	—	—	—	—	—	—	—	—	—	–
	ADDX, SUBX	B	B	—	—	—	—	—	—	—	—	—	–
	ADDS, SUBS	—	L	—	—	—	—	—	—	—	—	—	–
	INC, DEC	—	BWL	—	—	—	—	—	—	—	—	—	–
	DAA, DAS	—	B	—	—	—	—	—	—	—	—	—	–
	MULXU, DIVXU	—	BW	—	—	—	—	—	—	—	—	—	–
	MULXS, DIVXS	—	BW	—	—	—	—	—	—	—	—	—	–
	NEG	—	BWL	—	—	—	—	—	—	—	—	—	–
	EXTU, EXTS	—	WL	—	—	—	—	—	—	—	—	—	–
	TAS*2	—	—	B	—	—	—	—	—	—	—	—	–
Logic operations	AND, OR, XOR	BWL	BWL	—	—	—	—	—	—	—	—	—	–
	NOT	—	BWL	—	—	—	—	—	—	—	—	—	–
Shift		—	BWL	—	—	—	—	—	—	—	—	—	–
Bit-manipulation		—	B	B	—	—	—	B	B	—	B	—	–
Branch	Bcc, BSR	—	—	—	—	—	—	—	—	—	—	○	○
	JMP, JSR	—	—	—	—	—	—	—	—	○	—	—	–
	RTS	—	—	—	—	—	—	—	—	—	—	—	–

	LDC	B	B	W	W	W	W	—	W	—	W	—	—
	STC	—	B	W	W	W	W	—	W	—	W	—	—
	ANDC, ORC, XORC	B	—	—	—	—	—	—	—	—	—	—	—
	NOP	—	—	—	—	—	—	—	—	—	—	—	—
Block data transfer		—	—	—	—	—	—	—	—	—	—	—	—

Legend:

B: Byte

W: Word

L: Longword

Notes: 1. Cannot be used in this LSI.

2. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS in

3. Only registers ER0 to ER6 should be used when using the STM/LDM instruc

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 r
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 @SP+, Rn. 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, @–S
LDM*3	L	@SP+ → Rn (register list) Pops two or more general registers from the s
STM*3	L	Rn (register list) → @–SP Pushes two or more general registers onto th

		in two general registers, or on immediate da in a general register.
INC DEC	B/W/L	Rd ± 1 → Rd, Rd ± 2 → Rd Increments or decrements a general register (Byte operands can be incremented or decre 1 only.)
ADDS SUBS	L	Rd ± 1 → Rd, Rd ± 2 → Rd, Rd ± 4 → Rd Adds or subtracts the value 1, 2, or 4 to or fr 32-bit register.
DAA DAS	B	Rd decimal adjust → Rd Decimal-adjusts an addition or subtraction re general register by referring to the CCR to p BCD data.
MULXU	B/W	Rd × Rs → Rd Performs unsigned multiplication on data in registers: either 8 bits × 8 bits → 16 bits or 1 16 bits → 32 bits.
MULXS	B/W	Rd × Rs → Rd Performs signed multiplication on data in two registers: either 8 bits × 8 bits → 16 bits or 1 16 bits → 32 bits.
DIVXU	B/W	Rd ÷ Rs → Rd Performs unsigned division on data in two g registers: either 16 bits ÷ 8 bits → 8-bit quoti remainder or 32 bits ÷ 16 bits → 16-bit quoti bit remainder.

		bits according to the result.
NEG	B/W/L	0 – Rd → Rd Takes the two's complement (arithmetic comp data in a general register.
EXTU	W/L	Rd (zero extension) → Rd Extends the lower 8 bits of a 16-bit register to or the lower 16 bits of a 32-bit register to long 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 or the lower 16 bits of a 32-bit register to long by extending the sign bit.
TAS	B	@ERd – 0, 1 → (<bit 7> of @ERd)*2 Tests memory contents, and sets the most sig (bit 7) to 1.

			register and another general register or imm
	NOT	B/W/L	¬ (Rd) → (Rd) Takes the one's complement (logical comple general register contents.
Shift operations	SHAL SHAR	B/W/L	Rd (shift) → Rd Performs an arithmetic shift on general regis A 1-bit or 2-bit shift is possible.
	SHLL SHLR	B/W/L	Rd (shift) → Rd Performs a logical shift on general register c A 1-bit or 2-bit shift is possible.
	ROTL ROTR	B/W/L	Rd (rotate) → Rd Rotates general register contents. 1-bit or 2-bit rotation is possible.
	ROTXL ROTXR	B/W/L	Rd (rotate) → Rd Rotates general register contents through th 1-bit or 2-bit rotation is possible.

		immediate data or the lower three bits of a ge register.
BNOT	B	¬ (<bit-No.> of <EAd>) → (<bit-No.> of <EAd Inverts a specified bit in a general register or operand. The bit number is specified by 3-bit data or the lower three bits of a general regist
BTST	B	¬ (<bit-No.> of <EAd>) → Z Tests a specified bit in a general register or m operand and sets or clears the Z flag accordi bit number is specified by 3-bit immediate dat lower three bits of a general register.
BAND	B	C ∧ (<bit-No.> of <EAd>) → C ANDs the carry flag with a specified bit in a g register or memory operand and stores the re carry flag.
BIAND	B	C ∧ ¬ (<bit-No.> of <EAd>) → C ANDs the carry flag with the inverse of a spec a general register or memory operand and sto result in the carry flag. The bit number is specified by 3-bit immediat
BOR	B	C ∨ (<bit-No.> of <EAd>) → C ORs the carry flag with a specified bit in a ge register or memory operand and stores the re carry flag.
BIOR	B	C ∨ ¬ (<bit-No.> of <EAd>) → C ORs the carry flag with the inverse of a specif general register or memory operand and store result in the carry flag. The bit number is specified by 3-bit immediat

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

		BLS	Low or same	C ∨ Z
		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 ⊕
		BLT	Less than	N ⊕
		BGT	Greater than	Z∨(N
		BLE	Less or equal	Z∨(N
	JMP	—	Branches unconditionally to a specified addre	
	BSR	—	Branches to a subroutine at a specified addre	
	JSR	—	Branches to a subroutine at a specified addre	
	RTS	—	Returns from a subroutine	

		between them and memory. The upper 8 bit
STC	B/W	CCR → (EAd), EXR → (EAd) Transfers CCR or EXR contents to a genera memory. Although CCR and EXR are 8-bit r word-size transfers are performed between t memory. The upper 8 bits are valid.
ANDC	B	CCR ∧ #IMM → CCR, EXR ∧ #IMM → EXR Logically ANDs the CCR or EXR contents w immediate data.
ORC	B	CCR ∨ #IMM → CCR, EXR ∨ #IMM → EXR Logically ORs the CCR or EXR contents wit data.
XORC	B	CCR ⊕ #IMM → CCR, EXR ⊕ #IMM → EXR Logically exclusive-ORs the CCR or EXR co immediate data.
NOP	—	PC + 2 → PC Only increments the program counter.

		Until R4 = 0 else next;	
			Block transfer instruction. Transfers the numb bytes specified by R4L or R4 from locations s the address indicated by ER5 to locations sta address indicated by ER6. After the transfer, instruction is executed.

Notes: 1. Size refers to the operand size.
B: Byte
W: Word
L: Longword
2. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS in
3. Only registers ER0 to ER6 should be used when using the STM/LDM instruc

2.6.4 Basic Instruction Formats

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

Operation Field: Indicates the function of the instruction, the addressing mode, and th to be carried out on the operand. The operation field always includes the first four bits o instruction. Some instructions have two operation fields.

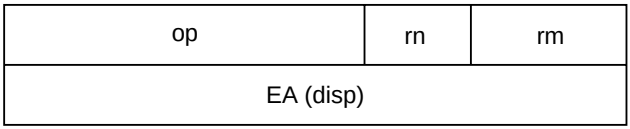
Register Field: Specifies a general register. Address registers are specified by 3 bits, d by 3 bits or 4 bits. Some instructions have two register fields. Some have no register fie

Effective Address Extension: Eight, 16, or 32 bits specifying immediate data, an abso address, or a displacement.

Condition Field: Specifies the branching condition of Bcc instructions.

Figure 2.12 shows examples of instruction formats.

(3) Operation field, register fields, and effective address extension



MOV.B @(d:16, R

(4) Operation field, effective address extension, and condition field

BRA d:16, etc

Figure 2.12 Instruction Formats (Examples)

2.6.5 Notes on Use of Bit-Manipulation Instructions

The BSET, BCLR, BNOT, BST, and BIST instructions read a byte of data, carry out b manipulation, then write back the byte of data. Caution is therefore required when usi instructions on a register containing write-only bits, or a port.

The BCLR instruction can be used to clear internal I/O register flags to 0. In this case, flag need not be read beforehand if it is clear that it has been set to 1 in an interrupt ha routine, etc.

2.7 Addressing Modes and Effective Address Calculation

2.7.1 Addressing Mode

The CPU supports the eight addressing modes listed in table 2.4. Each instruction use these addressing modes. Arithmetic and logic instructions can use the register direct a immediate modes. Data transfer instructions can use all addressing modes except prog relative and memory indirect. Bit-manipulation instructions use register direct, registe

3	Register indirect with displacement	@(d:16,ERn)/@(d:32,ERn)
4	Register indirect with post-increment Register indirect with pre-decrement	@ERn+ @–ERn
5	Absolute address	@aa:8/@aa:16/@aa:24/@a
6	Immediate	#xx:8/#xx:16/#xx:32
7	Program-counter relative	@(d:8,PC)/@(d:16,PC)
8	Memory indirect	@@aa:8

Register Direct—Rn

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

Register Indirect—@ERn

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

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

A 16-bit or 32-bit displacement contained in the instruction is added to an address regis
specified by the register field of the instruction, and the sum gives the address of a mem
operand. A 16-bit displacement is sign-extended when added.

- Register indirect with pre-decrement—@-ERn

  The value 1, 2, or 4 is subtracted from an address register (ERn) specified by the r
  in the instruction code, and the result becomes the address of a memory operand. T
  also stored in the address register. The value subtracted is 1 for byte access, 2 for w
  or 4 for longword access. For word or longword access, the register value should b

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

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

To access data, the absolute address should be 8 bits (@aa:8), 16 bits (@aa:16), or 32
(@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. A 32-bit absolute
access the entire address space.

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

Table 2.5 indicates the accessible absolute address ranges.

Table 2.5 Absolute Address Access Ranges

Absolute Address		Normal Mode	Advanced Mod
Data address	8 bits (@aa:8)	H'FF00 to H'FFFF	H'FFFF00 to H'F
	16 bits (@aa:16)	H'0000 to H'FFFF	H'000000 to H'0 H'FF8000 to H'F
	32 bits (@aa:32)		H'000000 to H'F
Program instruction address	24 bits (@aa:24)		

Program-Counter Relative—@(d:8, PC) or @(d:16, PC)

This mode is used in the Bcc and BSR instructions. An 8-bit or 16-bit displacement cor
the instruction is sign-extended and added to the 24-bit PC contents to generate a branc
Only the lower 24 bits of this branch address are valid; the upper 8 bits are all assumed
(H'00). The PC value to which the displacement is added is the address of the first byte
instruction, so the possible branching range is –126 to +128 bytes (–63 to +64 words) o
+32768 bytes (–16383 to +16384 words) from the branch instruction. The resulting val
be an even number.

Memory Indirect—@@aa:8

This mode can be used by the JMP and JSR instructions. The instruction code contains
absolute address specifying a memory operand. This memory operand contains a branc
The upper bits of the absolute address are all assumed to be 0, so the address range is 0
(H'0000 to H'00FF in normal mode, H'000000 to H'0000FF in advanced mode). In norm
the memory operand is a word operand and the branch address is 16 bits long. In advan
the memory operand is a longword operand, the first byte of which is assumed to be all

Note that the first part of the address range is also the exception vector area. For further
refer to section 4, Exception Handling.

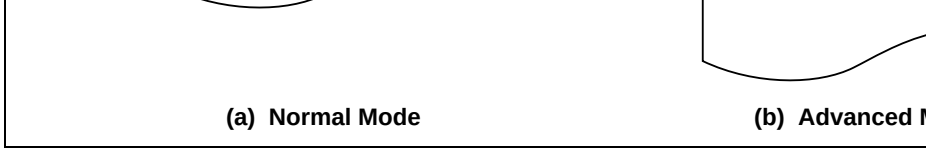
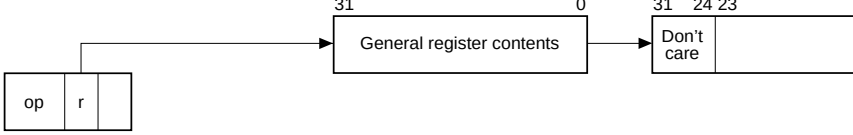


Figure 2.13 Branch Address Specification in Memory Indirect Mode

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

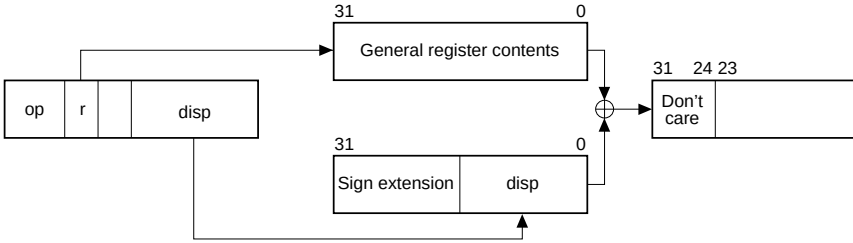
2.7.2 Effective Address Calculation

Table 2.6 indicates how effective addresses are calculated in each addressing mode. In mode the upper 8 bits of the effective address are ignored in order to generate a 16-bit



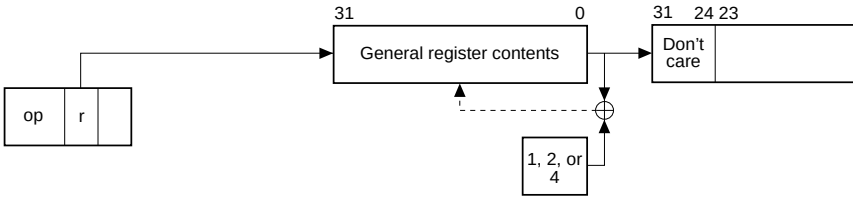
3 Register indirect with displacement

@(d:16, ERn) or @(d:32, ERn)

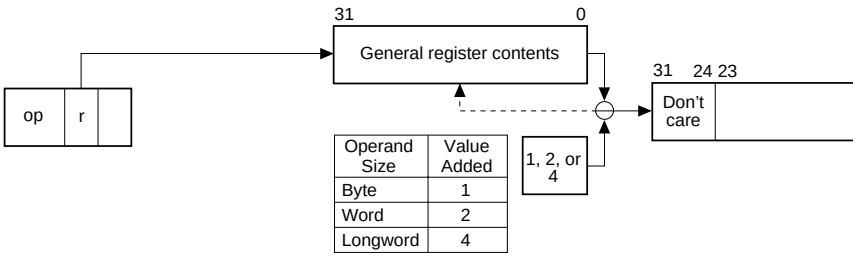


4 Register indirect with post-increment or pre-decrement

- Register indirect with post-increment @ERn+



- Register indirect with pre-decrement @–ERn

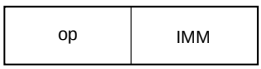


Operand Size	Value Added
Byte	1
Word	2
Longword	4

@aa:24

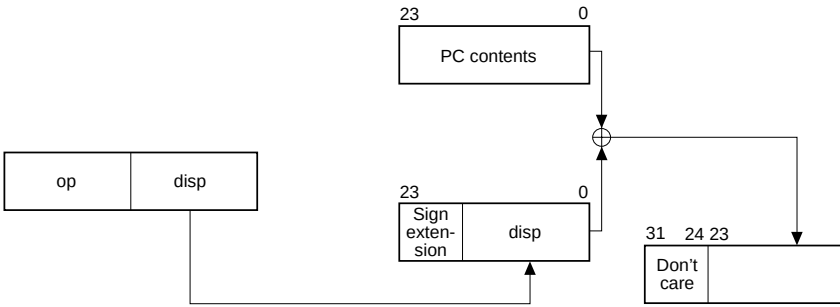
@aa:32

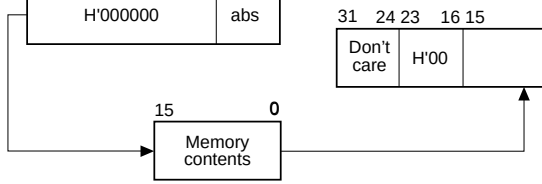
6 Immediate #xx:8/#xx:16/#xx:32 Operand is imm



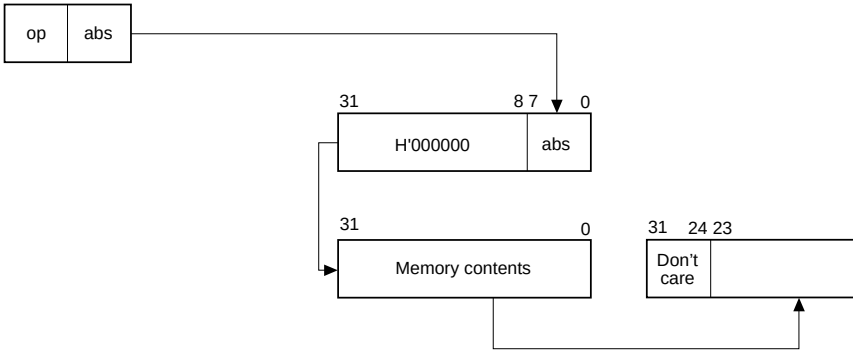
7 Program-counter relative

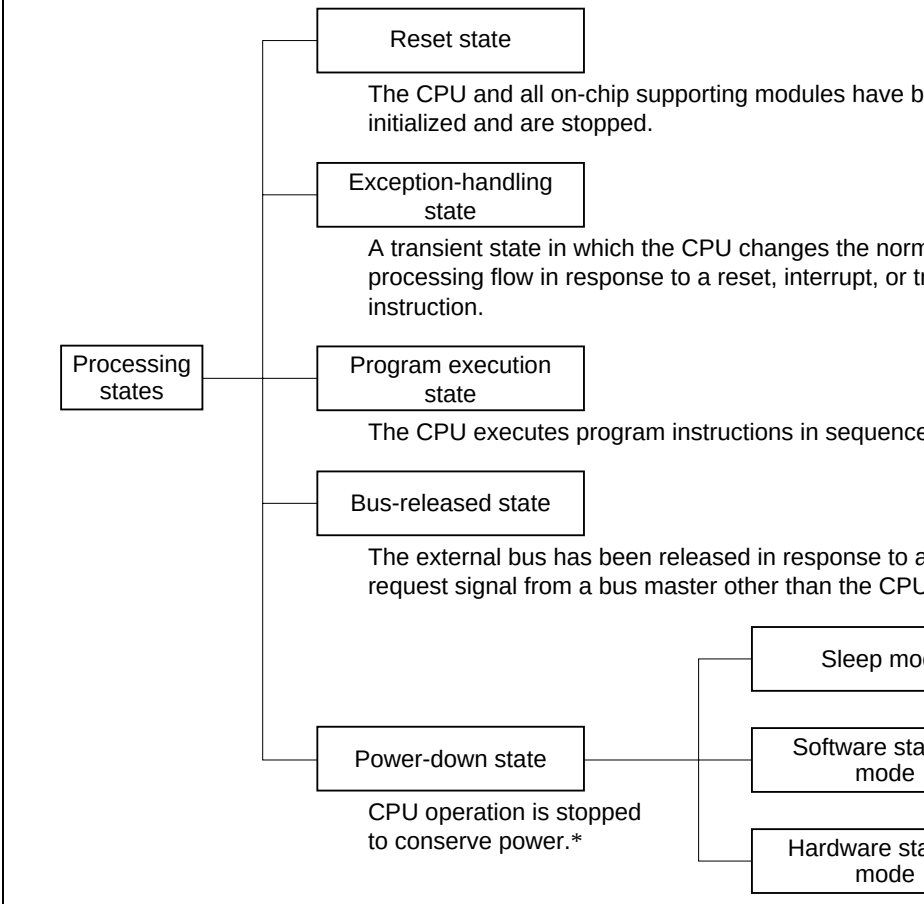
@(d:8, PC)/@(d:16, PC)





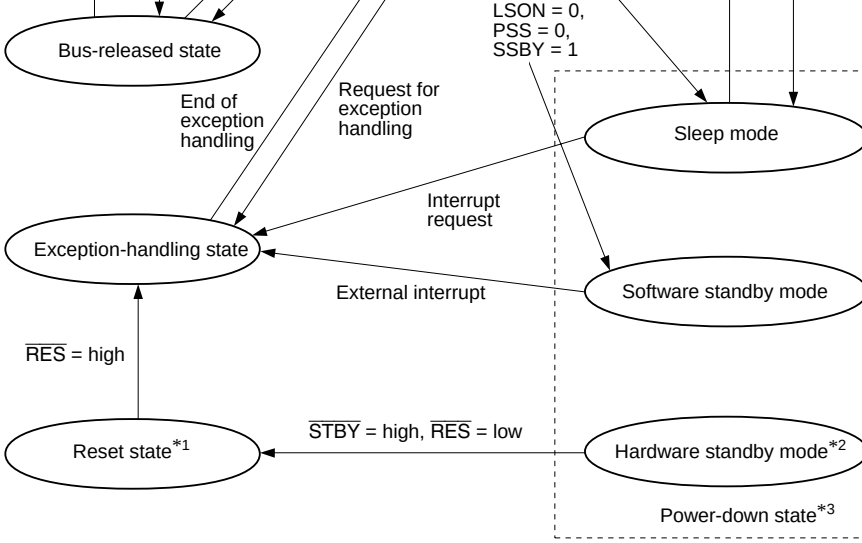
- Advanced mode





Note: * The power-down state also includes a medium-speed mode, module stop
sub-active mode, sub-sleep mode, and watch mode.

Figure 2.14 Processing States



Notes: 1. From any state except hardware standby mode, a transition to the reset state occurs whene goes low. A transition can also be made to the reset state when the watchdog timer overflow
2. From any state, a transition to hardware standby mode occurs when STBY goes low.
3. The power-down state also includes a watch mode, subactive mode, subsleep mode, etc. F refer to section 25, Power-Down State.

Figure 2.15 State Transitions

2.8.2 Reset State

When the RES input goes low all current processing stops and the CPU enters the reset interrupts are disabled in the reset state. Reset exception handling starts when the RES changes from low to high.

The reset state can also be entered by a watchdog timer overflow. For details, refer to s Watchdog Timer (WDT).

Exception handling is performed for resets, interrupts, and trap instructions. Table 2.7 the types of exception handling and their priority. Trap instruction exception handling accepted in the program execution state.

Exception handling and the stack structure depend on the interrupt control mode set in

Table 2.7 Exception Handling Types and Priority

Priority	Type of Exception	Detection Timing	Start of Exception
High ▲	Reset	Synchronized with clock	Exception handling immediately after a transition at the $\overline{\text{RES}}$ when the watchdog overflows.
	Interrupt	End of instruction execution or end of exception-handling sequence*1	When an interrupt is exception handling end of the current in current exception-ha sequence.
	Trap instruction	When TRAPA instruction is executed	Exception handling a trap (TRAPA) inst executed.*2
Low			

Notes: 1. Interrupts are not detected at the end of the ANDC, ORC, XORC, and LDC or immediately after reset exception handling.
2. Trap instruction exception handling is always accepted in the program exec

Reset Exception Handling

After the $\overline{\text{RES}}$ pin has gone low and the reset state has been entered, when $\overline{\text{RES}}$ goes h reset exception handling starts. When reset exception handling starts the CPU fetches address (vector) from the exception vector table and starts program execution from tha All interrupts, including NMI, are disabled during reset exception handling and after i

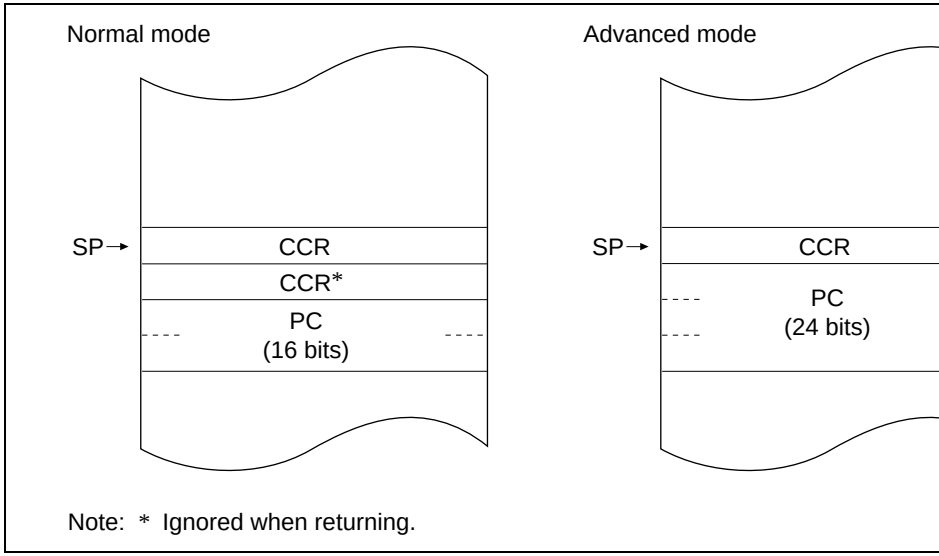


Figure 2.16 Stack Structure after Exception Handling (Examples)

2.8.4 Program Execution State

In this state the CPU executes program instructions in sequence.

2.8.5 Bus-Released State

This is a state in which the bus has been released in response to a bus request from a bu other than the CPU. While the bus is released, all CPU internal operations are halted.

There is one other bus master in addition to the CPU: the data transfer controller (DTC)

For further details, refer to section 6, Bus Controller.

mode, subsleep mode, and watch mode are power-down modes that use subclock inpu
details, refer to section 25, Power-Down State.

Sleep Mode

A transition to sleep mode is made if the SLEEP instruction is executed while the soft
standby bit (SSBY) in the standby control register (SBYCR) and the LSON bit in the
control register (LPWRCR) are both cleared to 0. In sleep mode, CPU operations stop
immediately after execution of the SLEEP instruction. The contents of CPU registers

Software Standby Mode

A transition to software standby mode is made if the SLEEP instruction is executed w
SSBY bit in SBYCR is set to 1 and the LSON bit in LPWRCR and the PSS bit in the
control/status register (TCSR) are both cleared to 0. In software standby mode, the CP
halt and all MCU operations stop. As long as a specified voltage is supplied, the conte
registers and on-chip RAM are retained. The I/O ports also remain in their existing sta

Hardware Standby Mode

A transition to hardware standby mode is made when the $\overline{\text{STBY}}$ pin goes low. In hard
standby mode, the CPU and clock halt and all MCU operations stop. The on-chip supp
modules are reset, but as long as a specified voltage is supplied, on-chip RAM conten
retained.

2.9.2 On-Chip Memory (ROM, RAM)

On-chip memory is accessed in one state. The data bus is 16 bits wide, permitting both
word transfer instruction. Figure 2.17 shows the on-chip memory access cycle. Figure 2
the pin states.

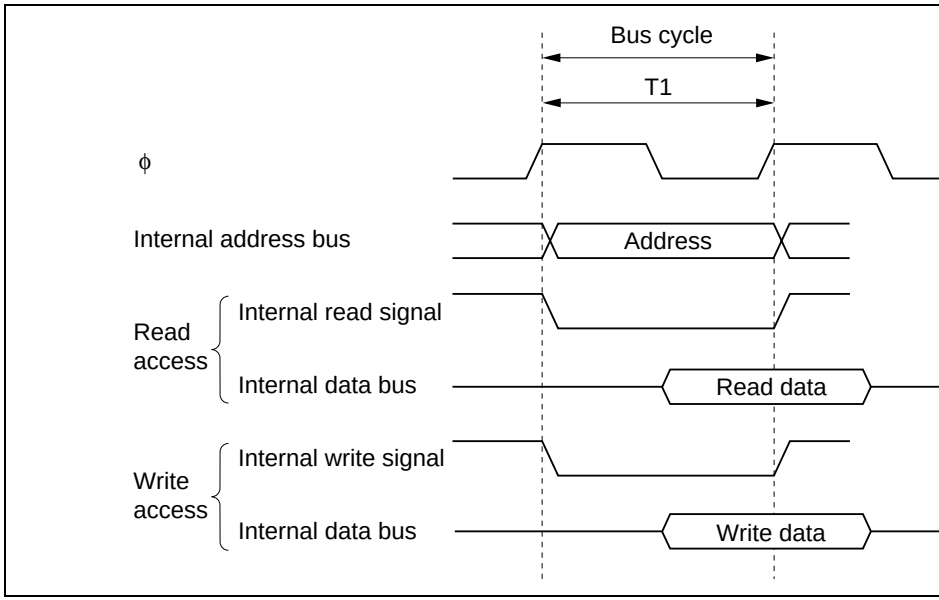


Figure 2.17 On-Chip Memory Access Cycle

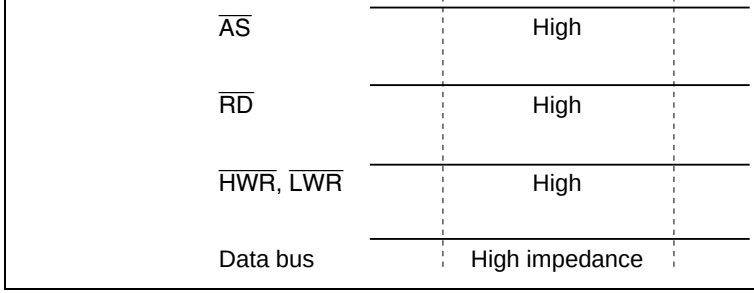


Figure 2.18 Pin States during On-Chip Memory Access

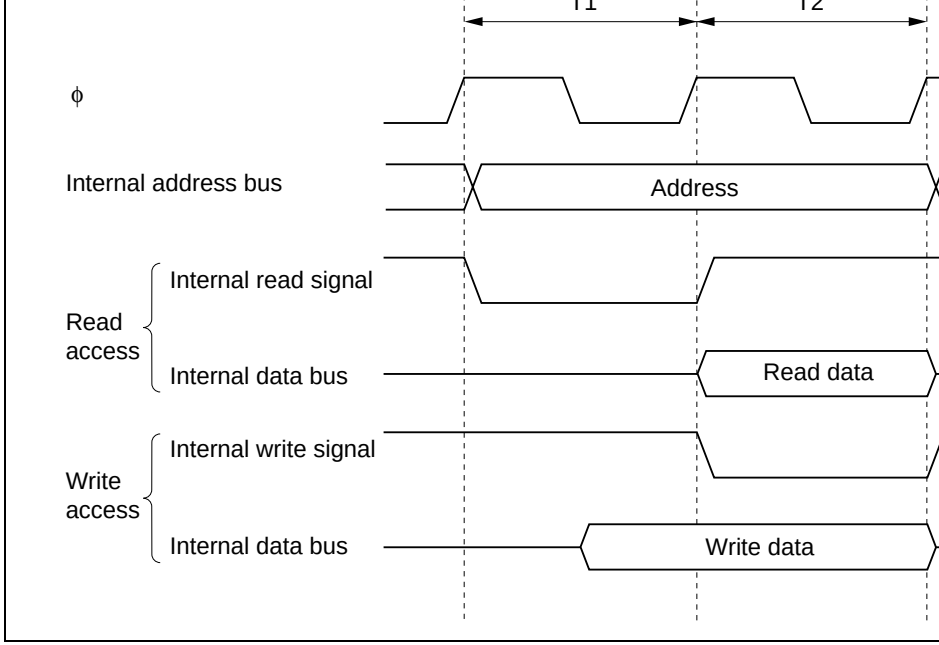


Figure 2.19 On-Chip Supporting Module Access Cycle

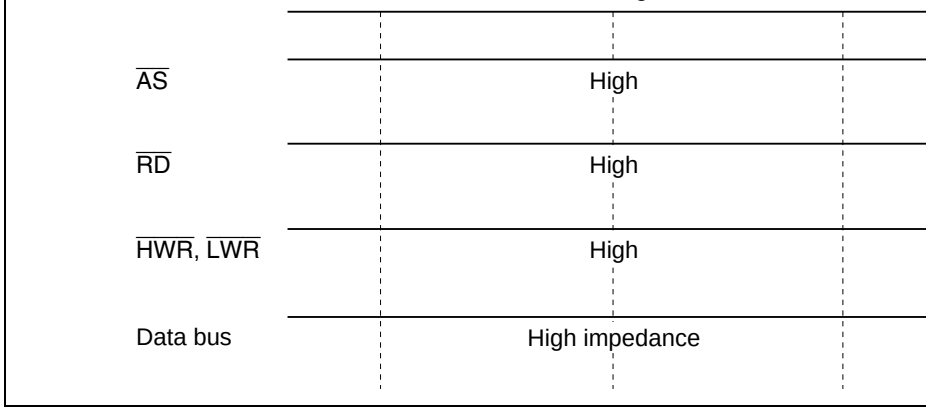


Figure 2.20 Pin States during On-Chip Supporting Module Access

2.9.4 External Address Space Access Timing

The external address space is accessed with an 8-bit or 16-bit data bus width in a two-
three-state bus cycle. In three-state access, wait states can be inserted. For further deta
section 6, Bus Controller.

2.10.2 STM/LDM Instruction

ER7 is not used as the register that can be saved (STM)/restored (LDM) when using ST
instruction, because ER7 is the stack pointer. Two, three, or four registers can be saved
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 H8S and H8
C/C++compilers.

operating mode and enabling/disabling of on-chip ROM, by setting the mode pins (MD1 and MD0).

Table 3.1 lists the MCU operating modes.

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

The CPU's architecture allows for 4 Gbytes of address space, but this LSI actually acc maximum of 16 Mbytes.

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

This LSI can only be used in modes 1 to 3. These means that the mode pins must selec these modes. Do not changes the inputs at the mode pins during operation.

Name	Abbreviation	R/W	Initial Value	Addr
Mode control register	MDCR	R/W	Undetermined	H'FF
System control register	SYSCR	R/W	H'09	H'FF
Bus control register	BCR	R/W	H'D7	H'FF
Serial/timer control register	STCR	R/W	H'00	H'FF

Note: * Lower 16 bits of the address.

3.2 Register Descriptions

3.2.1 Mode Control Register (MDCR)

Bit	7	6	5	4	3	2	1
	EXPE	—	—	—	—	—	MDS1
Initial value	—*	0	0	0	0	0	—*
Read/Write	R/W*	—	—	—	—	—	R

Note: * Determined by pins MD1 and MD0.

MDCR is an 8-bit read-only register that indicates the operating mode setting and the c
operating mode of the MCU.

The EXPE bit is initialized in coordination with the mode pin states by a reset and in ha
standby mode.

Bits 6 to 2—Reserved: These bits cannot be modified and are always read as 0.

Bits 1 and 0—Mode Select 1 and 0 (MDS1, MDS0): These bits indicate the input le
MD1 and MD0 (the current operating mode). Bits MDS1 and MDS0 correspond to M
MD0. MDS1 and MDS0 are read-only bits—they cannot be written to. The mode pin
MD0) input levels are latched into these bits when MDCR is read.

3.2.2 System Control Register (SYSCR)

Bit	7	6	5	4	3	2	1
	CS2E	IOSE	INTM1	INTM0	XRST	NMIEG	HIE
Initial value	0	0	0	0	1	0	0
Read/Write	R/W	R/W	R	R/W	R	R/W	R/W

SYSCR is an 8-bit readable/writable register that performs selection of system pin fun
source monitoring, interrupt control mode selection, NMI detected edge selection, sup
module pin location selection, supporting module register access control, and RAM ac
control.

Only bits 7, 6, 3, 1, and 0 are described here. For a detailed description of these bits, r
the description of the relevant modules (host interface, bus controller, watchdog timer
etc.). For information on bits 5, 4, and 2, see section 5.2.1, System Control Register (S

SYSCR is initialized to H'09 by a reset and in hardware standby mode. It is not initiali
software standby mode.

Bit 7—Chip Select 2 Enable (CS2E): Specifies the location of the host interface con
($\overline{CS2}$). For details, see section 18, Host Interface. The H8S/2144 Group does not incor
host interface, so do not set this bit to 1 in the H8S/2144 Group.

Note: * In the H8S/2148 F-ZTAT A-mask version and H8S/2147 F-ZTAT A-mask ver address range is from H'(FF)F000 to H'(FF)F7FF.

Bit 3—External Reset (XRST): Indicates the reset source. When the watchdog timer i reset can be generated by watchdog timer overflow as well as by external reset input. X read-only bit. It is set to 1 by an external reset and cleared to 0 by watchdog timer over

Bit 3

XRST	Description	
0	A reset is generated by watchdog timer overflow	
1	A reset is generated by an external reset	(I

Bit 1—Host Interface Enable (HIE): This bit controls CPU access to the host interfac registers and control registers (HICR, IDR1, ODR1, STR1, IDR2, ODR2, and STR2), t keyboard controller and MOS input pull-up control registers (KMIMR, KMPCR, and K the 8-bit timer (channel X and Y) data registers and control registers (TCRX/TCRY, TCSRX/TCSRY, TICRR/TCORAY, TICRF/TCORBY, TCNTX/TCNTY, TCORC/TI TCORAX, and TCORBX), and the timer connection control registers (TCONRI, TCO TCONRS, and SEDGR).

Bit 1

HIE	Description	
0	In areas H'(FF)FFF0 to H'(FF)FFF7 and H'(FF)FFFC to H'(FF)FFFF, CPU access to 8-bit timer (channel X and Y) data registers and control registers, and timer connection control registers, is permitted	(I
1	In areas H'(FF)FFF0 to H'(FF)FFF7 and H'(FF)FFFC to H'(FF)FFFF, CPU access to host interface data registers and control registers, and keyboard controller and MOS input pull-up control registers, is permitted	

3.2.3 Bus Control Register (BCR)

Bit	7	6	5	4	3	2	1
	ICIS1	ICIS0	BRSTRM	BRSTS1	BRSTS0	—	IOS1
Initial value	1	1	0	1	0	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BCR is an 8-bit readable/writable register that specifies the external memory space ac
and the I/O area range when the $\overline{AS}$ pin is designated for use as the I/O strobe. For det
to 2, see section 6.2.1, Bus Control Register (BCR).

BCR is initialized to H'D7 by a reset and in hardware standby mode.

Bits 1 and 0—IOS Select 1 and 0 (IOS1, IOS0): These bits specify the addresses for
$\overline{AS}$/$\overline{IOS}$ pin output goes low when IOSE = 1.

	BCR	
Bit 1	**Bit 0**	
IOS1	**IOS0**	**Description**
0	0	The $\overline{AS}$/$\overline{IOS}$ pin output goes low in accesses to addresses H'(FF)F000 to H'(FF)F03F
	1	The $\overline{AS}$/$\overline{IOS}$ pin output goes low in accesses to addresses H'(FF)F000 to H'(FF)F0FF
1	0	The $\overline{AS}$/$\overline{IOS}$ pin output goes low in accesses to addresses H'(FF)F000 to H'(FF)F3FF
	1	The $\overline{AS}$/$\overline{IOS}$ pin output goes low in accesses to addresses H'(FF)F000 to H'(FF)FE4F* (

Note: * In the H8S/2148 F-ZTAT A-mask version and H8S/2147 F-ZTAT A-mask v
address range is from H'(FF)F000 to H'(FF)F7FF.

(when the on-chip IIC option is included), an on-chip flash memory control (in F-ZTA
and also selects the TCNT input clock. For details of functions other than register acce
see the descriptions of the relevant modules. If a module controlled by STCR is not us
write 1 to the corresponding bit.

STCR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 5—I²C Control (IICS, IICX1, IICX0): These bits control the bus buffer fun
port A and the operation of the I²C bus interface when the on-chip IIC option is include
details, see section 16.2.7, Serial/Timer Control Register (STCR).

Bit 4—I²C Master Enable (IICE): Controls CPU access to the I²C bus interface data r
and control registers (ICCR, ICSR, ICDR/SARX, and ICMR/SAR), the PWMX data re
control registers (DADRAH/DACR, DADRAL, DADRBH/DACNTH, and DADRBL/
and the SCI control registers (SMR, BRR, and SCMR).

Bit 4

IICE	Description
0	Addresses H'(FF)FF88 and H'(FF)FF89, and H'(FF)FF8E and H'(FF)FF8F, for SCI1 control register access (I
	Addresses H'(FF)FFA0 and H'(FF)FFA1, and H'(FF)FFA6 and H'(FF)FFA7, for SCI2 control register access
	Addresses H'(FF)FFD8 and H'(FF)FFD9, and H'(FF)FFDE and H'(FF)FFDF for SCI0 control register access
1	Addresses H'(FF)FF88 and H'(FF)FF89, and H'(FF)FF8E and H'(FF)FF8F, for IIC1 data register and control register access
	Addresses H'(FF)FFA0 and H'(FF)FFA1, and H'(FF)FFA6 and H'(FF)FFA7, for PWMX data register and control register access
	Addresses H'(FF)FFD8 and H'(FF)FFD9, and H'(FF)FFDE and H'(FF)FFDF for IIC0 data register and control register access

	and supporting module control register access
1	Addresses H'(FF)FF80 to H'(FF)FF87 are used for flash memory control r access (F-ZTAT version only)

Bit 2—Reserved: Do not write 1 to this bit.

Bits 1 and 0—Internal Clock Select 1 and 0 (ICKS1, ICKS0): These bits, together CKS2 to CKS0 in TCR, select the clock to be input to TCNT. For details, see section Timer Control Register (TCR).

3.3 Operating Mode Descriptions

3.3.1 Mode 1

The CPU can access a 64-kbyte address space in normal mode. The on-chip ROM is d

Ports 1 and 2 function as an address bus, port 3 functions as a data bus, and part of por bus control signals. Clearing the ABW bit to 0 in the WSCR register makes port B a d

3.3.2 Mode 2

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM After a reset, single-chip mode is set, and the EXPE bit in MDCR must be set to 1 in o external addresses.

When the EXPE bit in MDCR is set to 1, ports 1, 2 and A function as input ports after They can be set to output addresses by setting the corresponding bits in the data direct (DDR) to 1. Port 3 functions as a data bus, and part of port 9 carries bus control signal the ABW bit to 0 in the WSCR register makes port B a data bus.

the ABW bit to 0 in the WSCR register makes port B a data bus.

In this operating mode, the available amount of on-chip ROM in products with 64 kbyt
of ROM is limited to 56 kbytes.

3.4 Pin Functions in Each Operating Mode

The pin functions of ports 1 to 3, 9, A, and B vary depending on the operating mode. T
shows their functions in each operating mode.

Table 3.3 Pin Functions in Each Mode

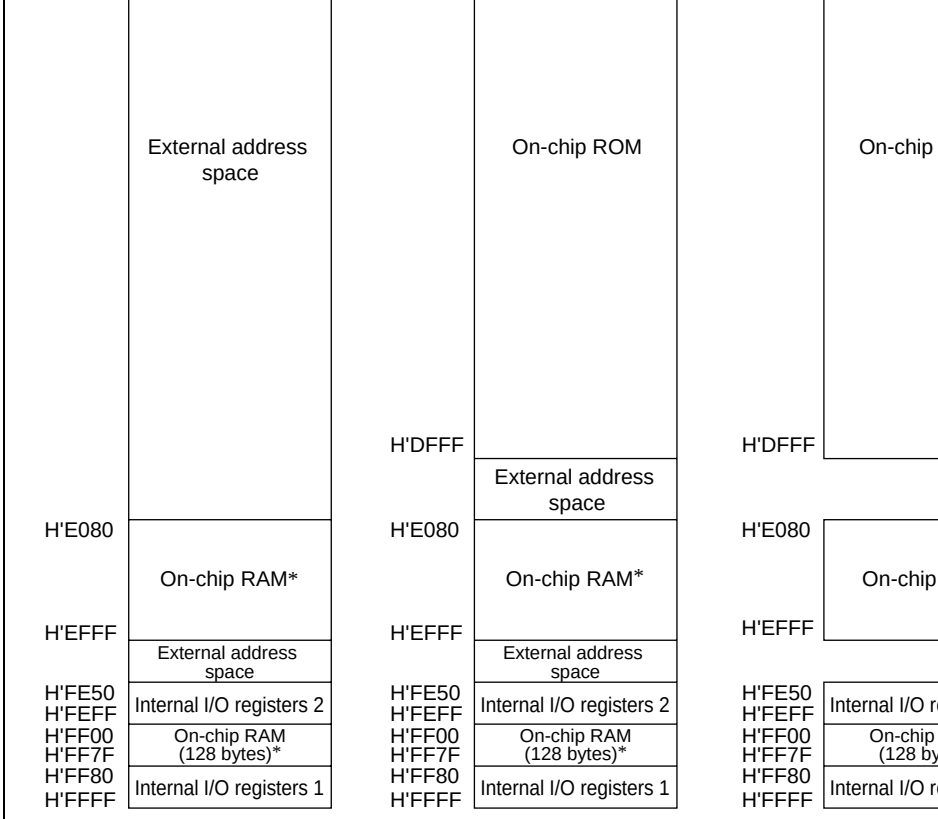
Port		Mode 1	Mode 2	Mode 3
Port 1		A	P*/A	P*/A
Port 2		A	P*/A	P*/A
Port A		P	P*/A	P
Port 3		D	P*/D	P*/D
Port B		P*/D	P*/D	P*/D
Port 9	P97	P*/C	P*/C	P*/C
	P96	C*/P	P*/C	P*/C
	P95 to P93	C	P*/C	P*/C
	P92 to P91	P	P	P
	P90	P*/C	P*/C	P*/C

Legend:
P: I/O port
A: Address bus output
D: Data bus I/O
C: Control signals, clock I/O
*: After reset

(normal mode).

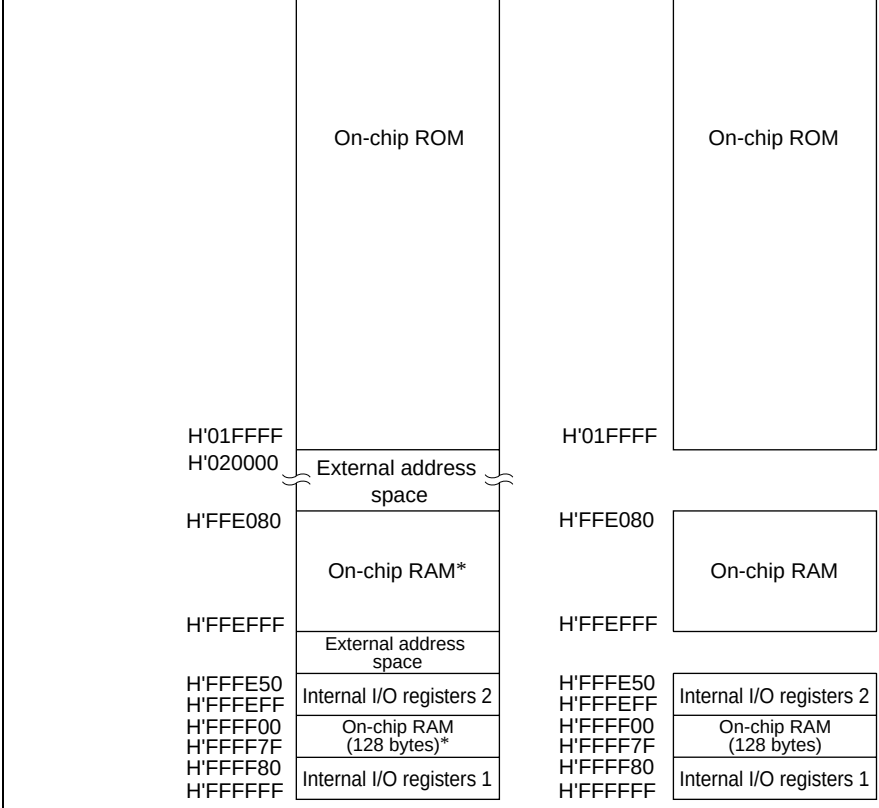
Do not access the reserved area and addresses of modules not supported by the produc
normal operation is not guaranteed when these regions are accessed.

For details, see section 6, Bus Controller.



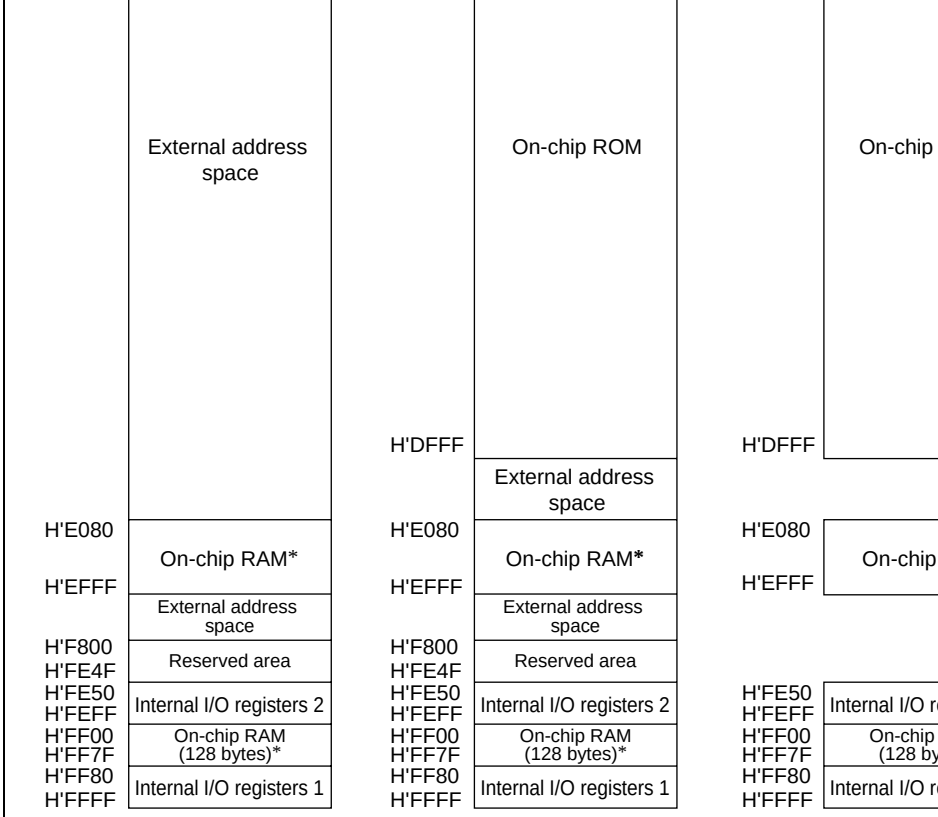
Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.1 H8S/2148 (Except for F-ZTAT A-Mask Version) and H8S/2144 Memo Each Operating Mode



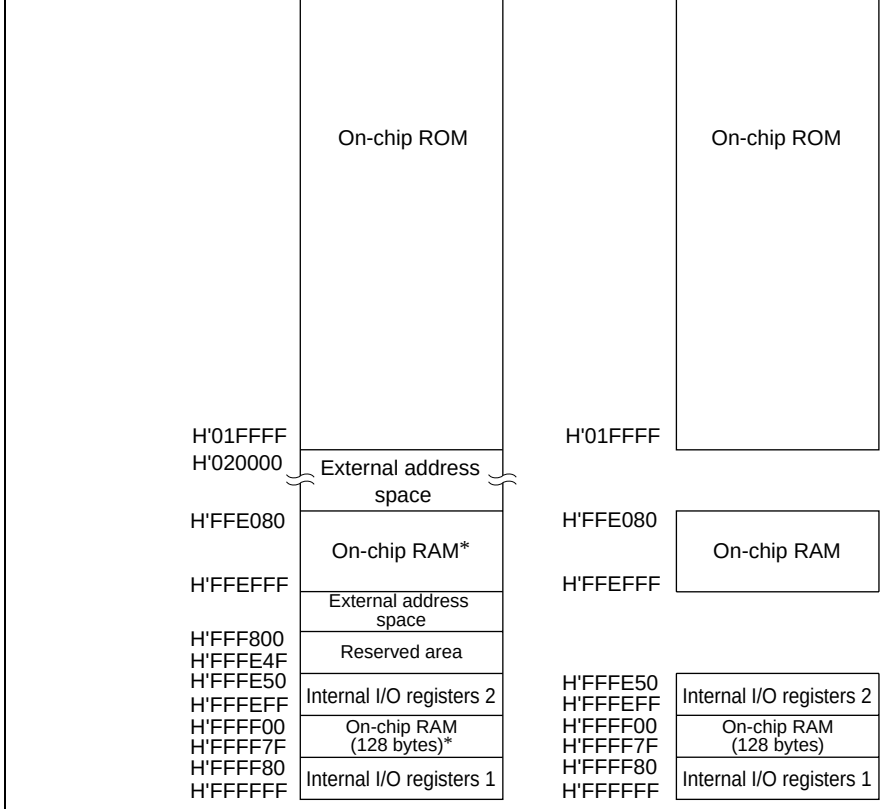
Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.1 H8S/2148 (Except for F-ZTAT A-Mask Version) and H8S/2144 Mem Each Operating Mode (cont)



Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.2 H8S/2148 F-ZTAT A-Mask Version Memory Map in Each Operati



Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.2 H8S/2148 F-ZTAT A-Mask Version Memory Map in Each Operat (cont)

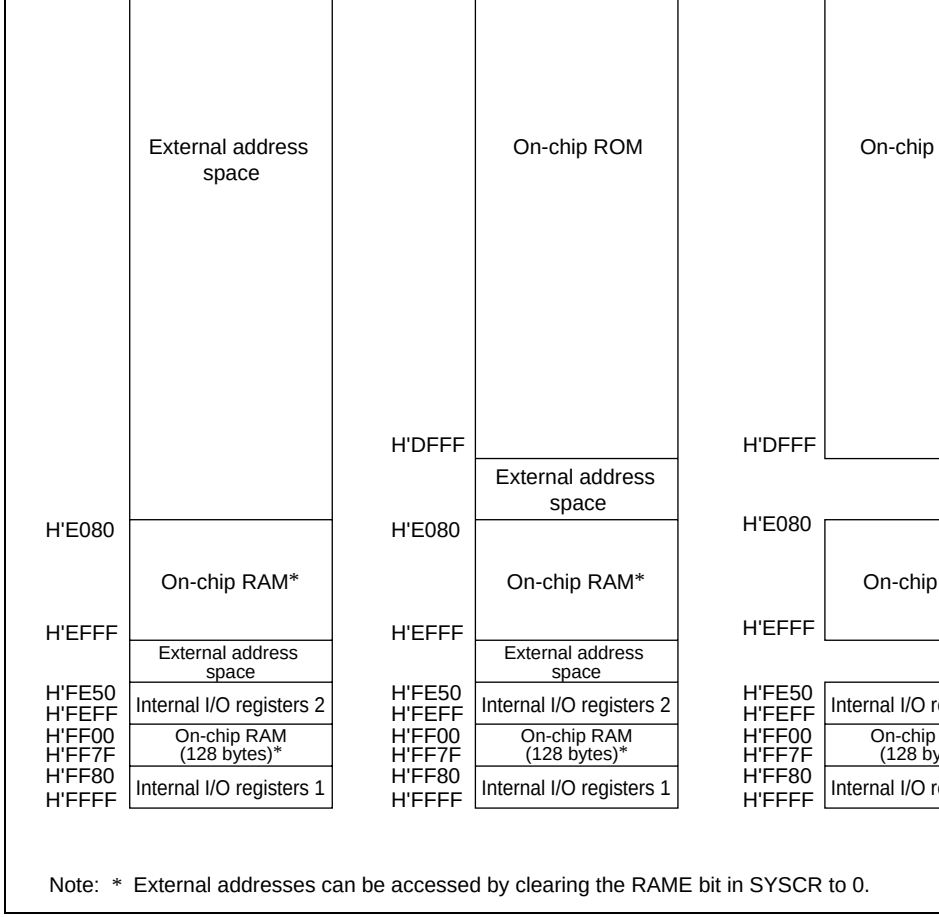
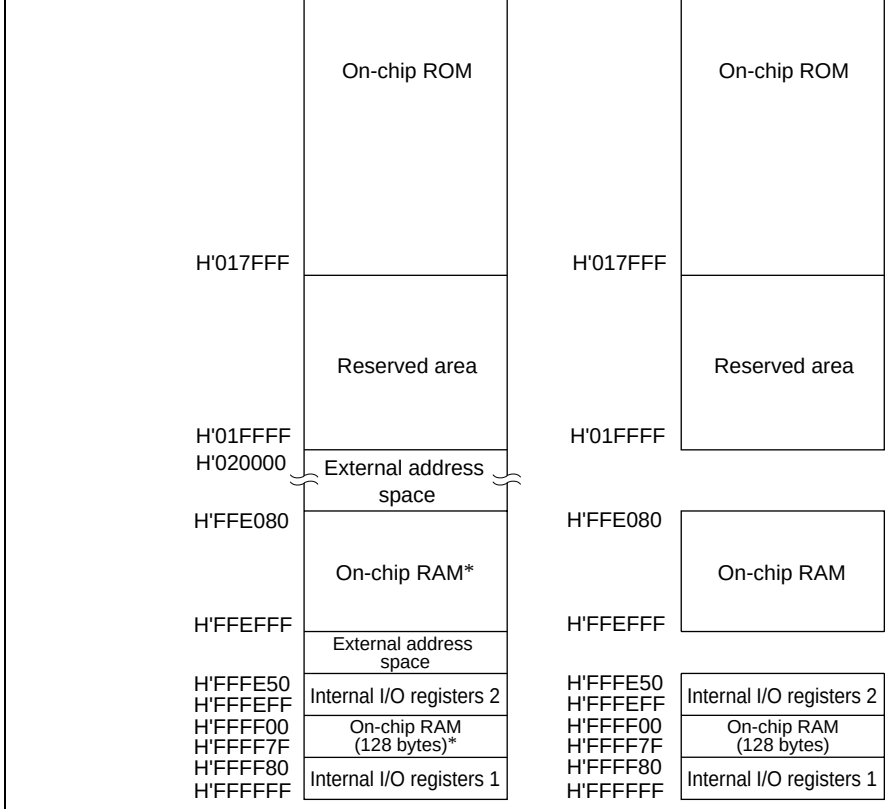
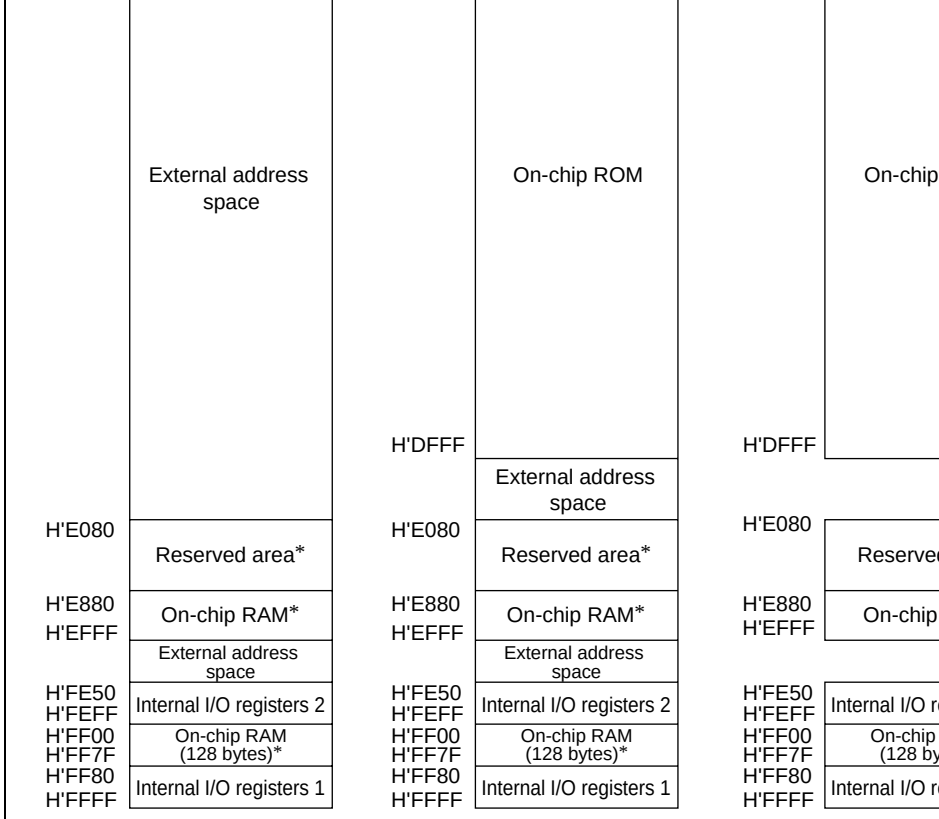


Figure 3.3 H8S/2143 Memory Map in Each Operating Mode



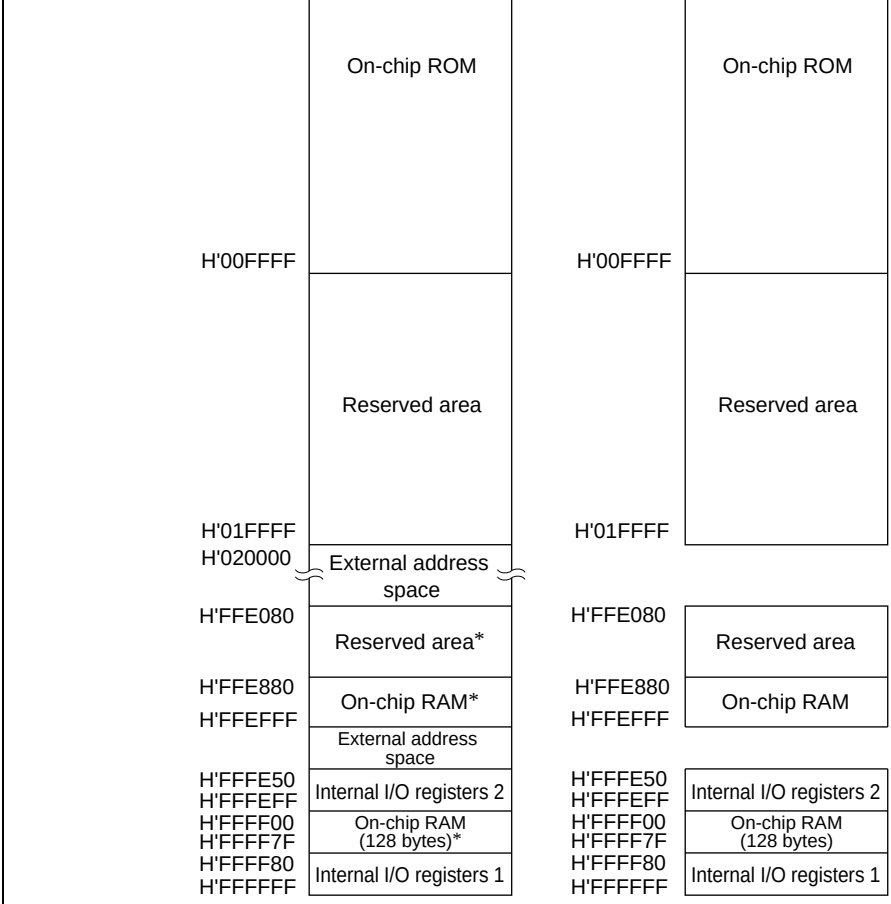
Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.3 H8S/2143 Memory Map in Each Operating Mode (cont)



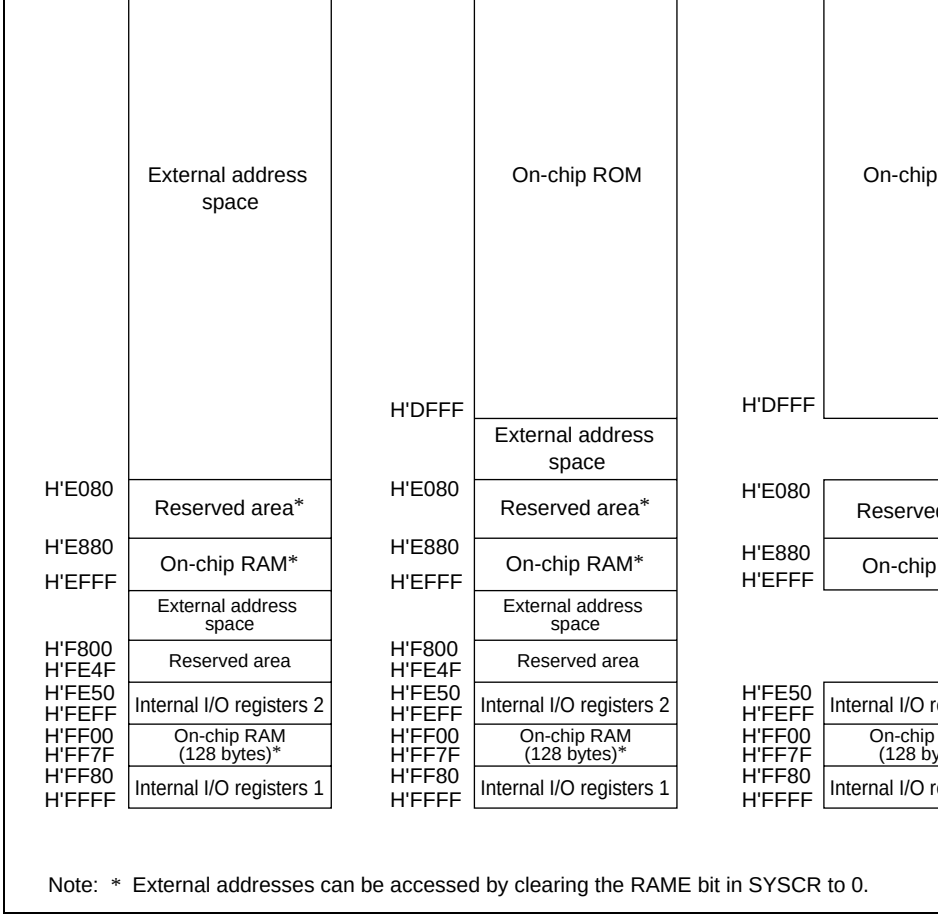
Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.4 H8S/2147 (Except for F-ZTAT A-Mask Version), H8S/2147N, and H
Memory Map in Each Operating Mode



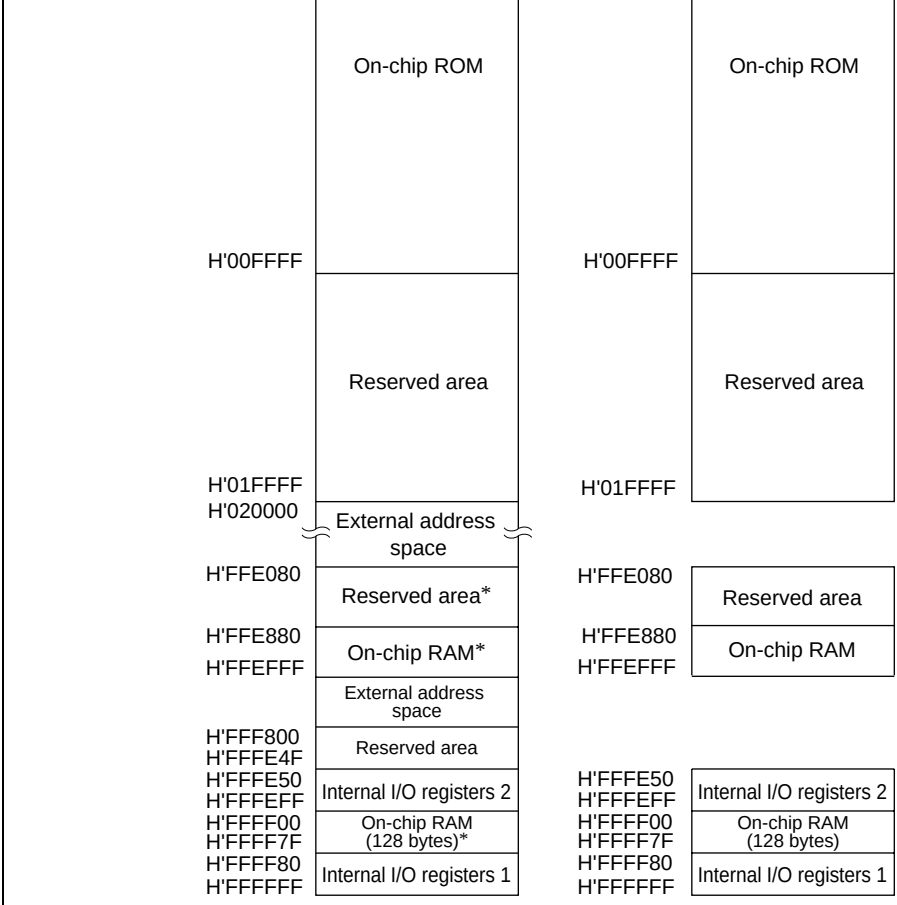
Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.4 H8S/2147 (Except for F-ZTAT A-Mask Version), H8S/2147N, and Memory Map in Each Operating Mode (cont)



Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.5 H8S/2147 F-ZTAT A-Mask Version Memory Map in Each Operati



Note: * External addresses can be accessed by clearing the RAME bit in SYSCR to 0.

Figure 3.5 H8S/2147 F-ZTAT A-Mask Version Memory Map in Each Operat (cont)

Exception handling is prioritized as shown in table 4.1. If two or more exceptions occu
simultaneously, they are accepted and processed in order of priority. Trap instruction e
are accepted at all times in the program execution state.

Exception handling sources, the stack structure, and the operation of the CPU vary dep
the interrupt control mode set by the INTM0 and INTM1 bits in SYSCR.

Table 4.1 Exception Types and Priority

Priority	Exception Type	Start of Exception Handling
High ▲	Reset	Starts immediately after a low-to-high transit $\overline{\text{RES}}$ pin, or when the watchdog timer overflo
	Trace*1	Starts when execution of the current instruct exception handling ends, if the trace (T) bit is
	Interrupt	Starts when execution of the current instruct exception handling ends, if an interrupt requ issued.*2
	Direct transition	Started by a direct transition resulting from e SLEEP instruction.
Low	Trap instruction (TRAPA)*3	Started by execution of a trap instruction (TR

Notes: 1. Traces are enabled only in interrupt control modes 2 and 3. (They cannot b this LSI.) Trace exception handling is not executed after execution of an RT instruction.
2. Interrupt detection is not performed on completion of ANDC, ORC, XORC, instruction execution, or on completion of reset exception handling.
3. Trap instruction exception handling requests are accepted at all times in the execution state.

For a reset exception, steps 2 and 3 above are carried out.

4.1.3 Exception Sources and Vector Table

The exception sources are classified as shown in figure 4.1. Different vector addresses a
assigned to different exception sources.

Table 4.2 lists the exception sources and their vector addresses.

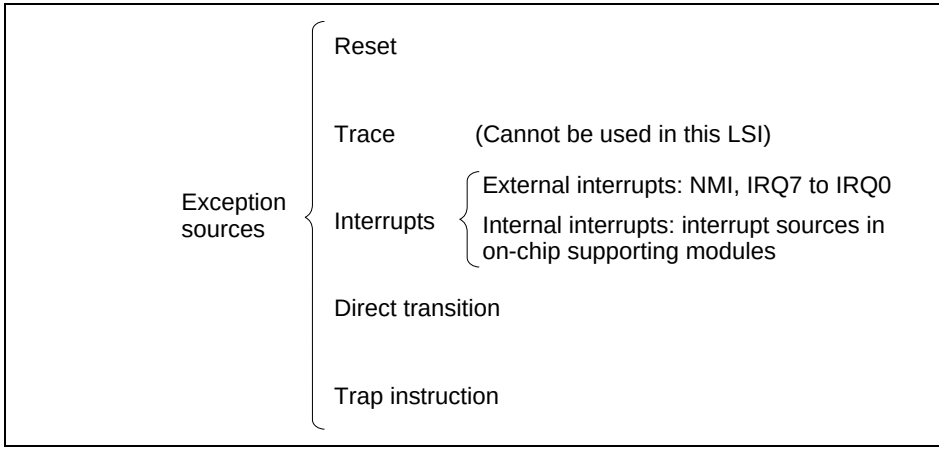


Figure 4.1 Exception Sources

		4	H'0008 to H'0009	H'0010 to
		5	H'000A to H'000B	H'0014 to
Direct transition		6	H'000C to H'000D	H'0018 to
External interrupt	NMI	7	H'000E to H'000F	H'001C to
Trap instruction (4 sources)		8	H'0010 to H'0011	H'0020 to
		9	H'0012 to H'0013	H'0024 to
		10	H'0014 to H'0015	H'0028 to
		11	H'0016 to H'0017	H'002C to
Reserved for system use		12	H'0018 to H'0019	H'0030 to
		13	H'001A to H'001B	H'0034 to
		14	H'001C to H'001D	H'0038 to
		15	H'001E to H'001F	H'003C to
External interrupt	IRQ0	16	H'0020 to H'0021	H'0040 to
	IRQ1	17	H'0022 to H'0023	H'0044 to
	IRQ2	18	H'0024 to H'0025	H'0048 to
	IRQ3	19	H'0026 to H'0027	H'004C to
	IRQ4	20	H'0028 to H'0029	H'0050 to
	IRQ5	21	H'002A to H'002B	H'0054 to
	IRQ6	22	H'002C to H'002D	H'0058 to
	IRQ7	23	H'002E to H'002F	H'005C to
Internal interrupt*2		24 | 103	H'0030 to H'0031 | H'00CE to H'00CF	H'0060 to | H'019C to

Notes: 1. Lower 16 bits of the address.

2. For details on internal interrupt vectors, see section 5.3.3, Interrupt Excepti Table.

Immediately after a reset, interrupt control mode 0 is set.

Reset exception handling begins when the $\overline{\text{RES}}$ pin changes from low to high.

The MCUs can also be reset by overflow of the watchdog timer. For details, see section Watchdog Timer (WDT).

4.2.2 Reset Sequence

The MCU enters the reset state when the $\overline{\text{RES}}$ pin goes low.

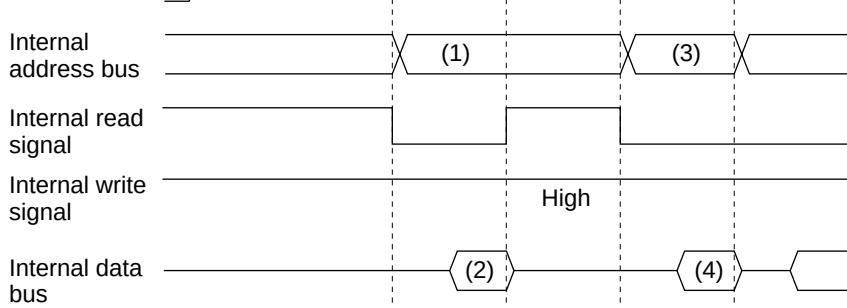
To ensure that the chip is reset, hold the $\overline{\text{RES}}$ pin low for at least 20 ms when powering reset the chip during operation, hold the $\overline{\text{RES}}$ pin low for at least 20 states. For pin state see appendix D.1, Port States in Each Processing State.

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

[1] The internal state of the CPU and the registers of the on-chip supporting modules ar initialized, and the I bit is set to 1 in CCR.

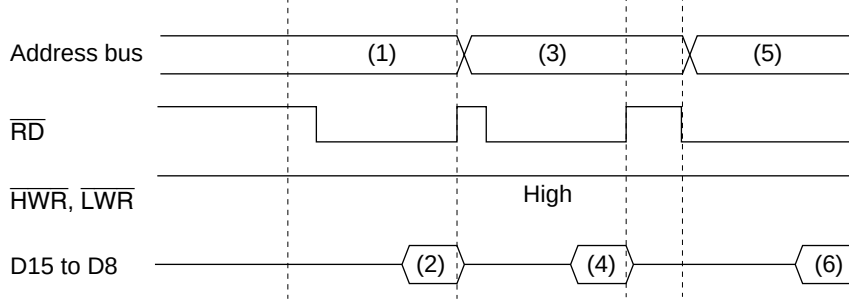
[2] The reset exception vector address is read and transferred to the PC, and program ex starts from the address indicated by the PC.

Figures 4.2 and 4.3 show examples of the reset sequence.



(1) Reset exception vector address ((1) = H'0000)
(2) Start address (contents of reset exception vector address)
(3) Start address ((3) = (2))
(4) First program instruction

Figure 4.2 Reset Sequence (Mode 3)



(1) (3) Reset exception vector address ((1) = H'0000, (3) = H'0001)
(2) (4) Start address (contents of reset exception vector address)
(5) Start address ((5) = (2) (4))
(6) First program instruction

Note: * 3 program wait states are inserted.

Figure 4.3 Reset Sequence (Mode 1)

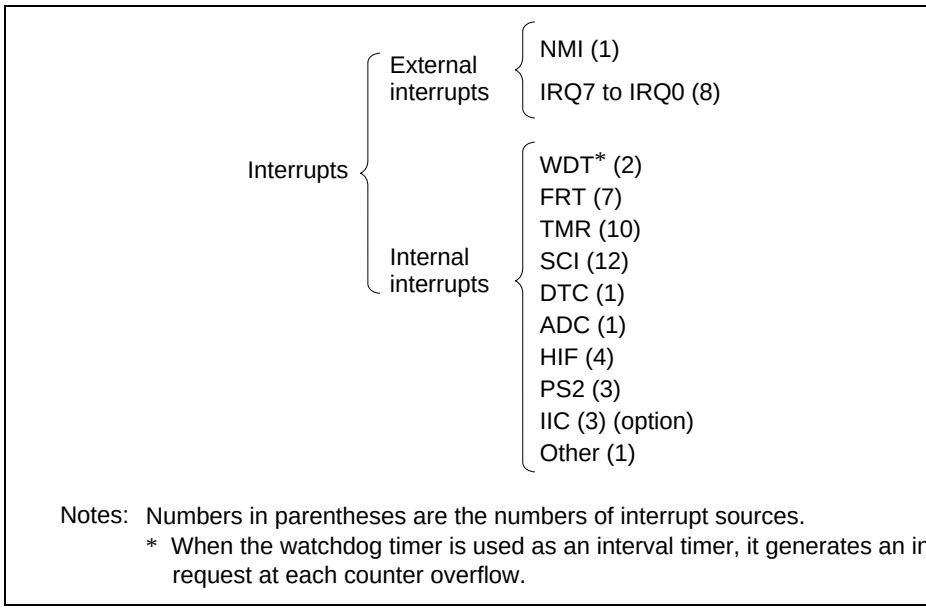
4.2.3 Interrupts after Reset

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

transfer controller (DTC), A/D converter (ADC), host interface (HIF), keyboard buffe
(PS2), and I^2C bus interface (option). Each interrupt source has a separate vector addr

NMI is the highest-priority interrupt. Interrupts are controlled by the interrupt controll
interrupt controller has two interrupt control modes and can assign interrupts other tha
address break to either three priority/mask levels to enable multiplexed interrupt contr

For details on interrupts, see section 5, Interrupt Controller.



Notes: Numbers in parentheses are the numbers of interrupt sources.
* When the watchdog timer is used as an interval timer, it generates an in
request at each counter overflow.

Figure 4.4 Interrupt Sources and Number of Interrupts

Table 4.3 Status of CCR and EXR after Trap Instruction Exception Handling

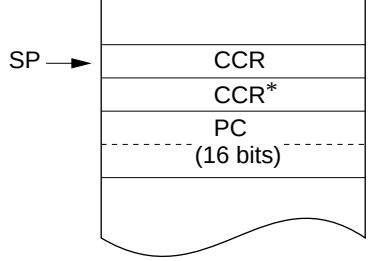
	CCR		EXR	
Interrupt Control Mode	**I**	**UI**	**I2 to I0**	**T**
0	1	—	—	—
1	1	1	—	—

Legend:

1: Set to 1

0: Cleared to 0

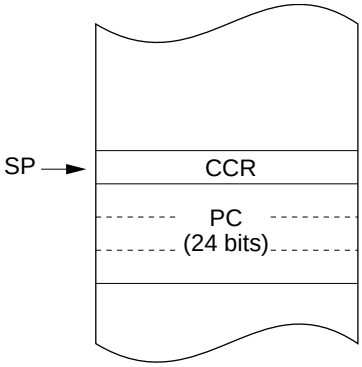
—: Retains value prior to execution.



Interrupt control modes 0 and 1

Note: * Ignored on return.

Figure 4.5 (1) Stack Status after Exception Handling (Normal Mode



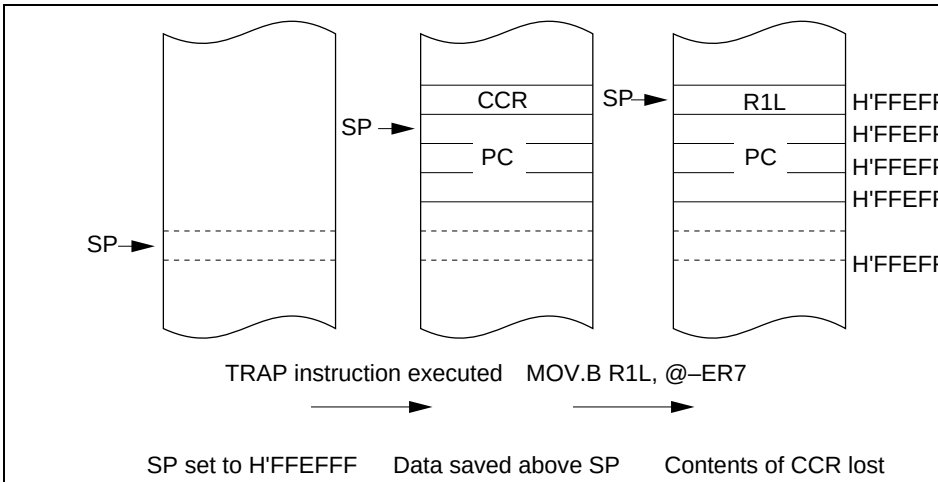
Interrupt control modes 0 and 1

Figure 4.5 (2) Stack Status after Exception Handling (Advanced Mod

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 4.6 shows an example of
happens when the SP value is odd.



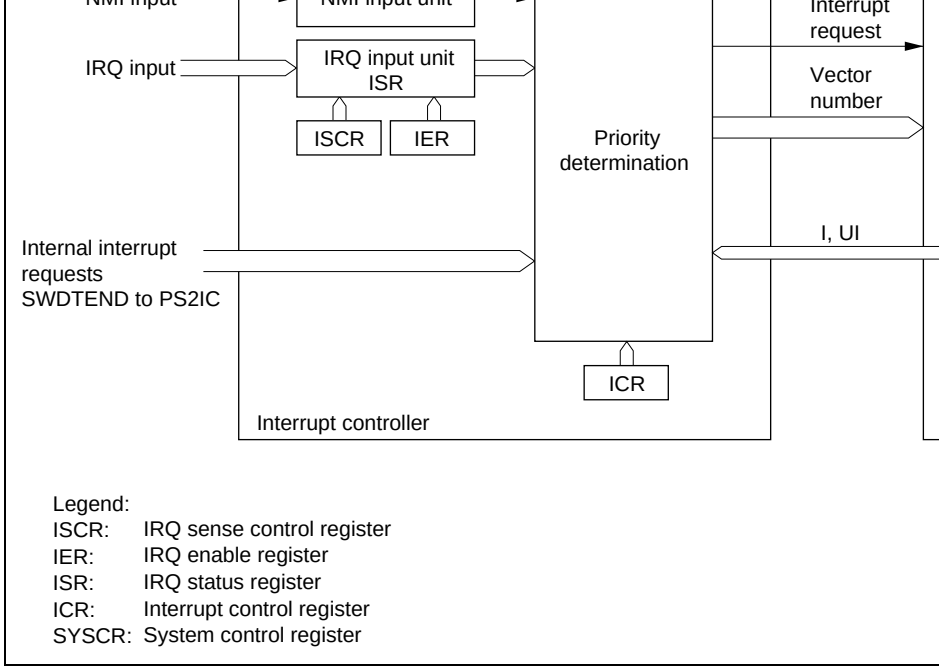
Legend:
CCR: Condition-code register
PC: Program counter
R1L: General register R1L
SP: Stack pointer

Note: This diagram illustrates an example in which the interrupt control mode is 0, in a
mode.

Figure 4.6 Operation when SP Value Is Odd

following features:

- Two interrupt control modes
  - Either of two interrupt control modes can be set by means of the INTM1 and I in the system control register (SYSCR).
- Priorities settable with ICR
  - An interrupt control register (ICR) is provided for setting interrupt priorities. T levels can be set for each module for all interrupts except NMI and address bre
- Independent vector addresses
  - All interrupt sources are assigned independent vector addresses, making it unn the source to be identified in the interrupt handling routine.
- Twenty-three external interrupt pins (nine external sources)
  - NMI is the highest-priority interrupt, and is accepted at all times. A rising or fa the NMI pin can be selected for the NMI interrupt.
  - Falling edge, rising edge, or both edge detection, or level sensing, at pins $\overline{\text{IRQ7}}$ can be selected for interrupts IRQ7 to IRQ0.
  - The IRQ6 interrupt is shared by the interrupt from the $\overline{\text{IRQ6}}$ pin and eight exte inputs ($\overline{\text{KIN7}}$ to $\overline{\text{KIN0}}$), and the IRQ7 interrupt is shared by the interrupt from and eight external interrupt inputs ($\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$). $\overline{\text{KIN15}}$ to $\overline{\text{KIN0}}$ can be m individually by the user program.
- DTC control
  - DTC activation is controlled by means of interrupts.



Legend:
ISCR: IRQ sense control register
IER: IRQ enable register
ISR: IRQ status register
ICR: Interrupt control register
SYSCR: System control register

Figure 5.1 Block Diagram of Interrupt Controller

External interrupt requests 7 to 0	$\overline{IRQ7}$ to $\overline{IRQ0}$	Input	Maskable external interrupts; risin both edges, or level sensing, can
Key input interrupt requests 15 to 0	$\overline{KIN15}$ to $\overline{KIN0}$	Input	Maskable external interrupts: fallin level sensing can be selected.

IRQ sense control register L	ISCRL	R/W	H'00	H'F
IRQ enable register	IER	R/W	H'00	H'F
IRQ status register	ISR	R/(W)*2	H'00	H'F
Keyboard matrix interrupt mask register	KMIMR	R/W	H'BF	H'F
Keyboard matrix interrupt mask register A	KMIMRA	R/W	H'FF	H'F
Interrupt control register A	ICRA	R/W	H'00	H'F
Interrupt control register B	ICRB	R/W	H'00	H'F
Interrupt control register C	ICRC	R/W	H'00	H'F
Address break control register	ABRKCR	R/W	H'00	H'F
Break address register A	BARA	R/W	H'00	H'F
Break address register B	BARB	R/W	H'00	H'F
Break address register C	BARC	R/W	H'00	H'F

Notes: 1. Lower 16 bits of the address.
2. Only 0 can be written, for flag clearing.
3. When setting KMIMR and KMIMRA, the HIE bit in SYSCR must be set to 1, MSTP2 bit in MSTPCRL must be set to 0.

Read/Write	R/W	R/W	R	R/W	R	R/W	R/W

SYSCR is an 8-bit readable/writable register that selects the interrupt control mode, a
detected edge for NMI, among other functions.

Only bits 5, 4, and 2 are described here; for details on the other bits, see section 3.2.2,
Control Register (SYSCR).

SYSCR is initialized to H'09 by a reset and in hardware standby mode. It is not initial
software standby mode.

Bits 5 and 4—Interrupt Control Mode 1 and 0 (INTM1, INTM0): These bits selec
interrupt control modes for the interrupt controller. The INTM1 bit must not be set to

Bit 5	Bit 4	Interrupt	
INTM1	**INTM0**	**Control Mode**	**Description**
0	0	0	Interrupts are controlled by I bit
	1	1	Interrupts are controlled by I and UI bits and ICI
1	0	2	Cannot be used in this LSI
	1	3	Cannot be used in this LSI

Bit 2—NMI Edge Select (NMIEG): Selects the input edge for the NMI pin.

Bit 2	
NMIEG	**Description**
0	Interrupt request generated at falling edge of NMI input
1	Interrupt request generated at rising edge of NMI input

interrupts other than NMI and address break.

The correspondence between ICR settings and interrupt sources is shown in table 5.3.

The ICR registers are initialized to H'00 by a reset and in hardware standby mode.

Bit n—Interrupt Control Level (ICRn): Sets the control level for the corresponding i source.

Bit n

ICRn	Description	
0	Corresponding interrupt source is control level 0 (non-priority)	(I
1	Corresponding interrupt source is control level 1 (priority)	

Note: n = 7 to 0

Table 5.3 Correspondence between Interrupt Sources and ICR Settings

	Bits						
Register	**7**	**6**	**5**	**4**	**3**	**2**	**1**
ICRA	IRQ0	IRQ1	IRQ2 IRQ3	IRQ4 IRQ5	IRQ6 IRQ7	DTC	Watchdog timer 0
ICRB	A/D converter	Free-running timer	—	—	8-bit timer channel 0	8-bit timer channel 1	8-bit timer channels X, Y
ICRC	SCI channel 0	SCI channel 1	SCI channel 2	IIC channel 0 (option)	IIC channel 1 (option)	—	—

IRQ7 to IRQ0.

IER is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 0—IRQ7 to IRQ0 Enable (IRQ7E to IRQ0E): These bits select whether I
IRQ0 are enabled or disabled.

Bit n

IRQnE	**Description**
0	IRQn interrupt disabled
1	IRQn interrupt enabled

Note: n = 7 to 0

5.2.4 IRQ Sense Control Registers H and L (ISCRH, ISCRL)

- ISCRH

Bit	15	14	13	12	11	10	9
	IRQ7SCB	IRQ7SCA	IRQ6SCB	IRQ6SCA	IRQ5SCB	IRQ5SCA	IRQ4SC
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ISCRL

Bit	7	6	5	4	3	2	1
	IRQ3SCB	IRQ3SCA	IRQ2SCB	IRQ2SCA	IRQ1SCB	IRQ1SCA	IRQ0SC
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ISCRL Bits 7 to 0		
IRQ7SCB to IRQ0SCB	**IRQ7SCA to IRQ0SCA**	**Description**
0	0	Interrupt request generated at $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ input low l
	1	Interrupt request generated at falling edge of $\overline{\text{IRQ7}}$ to $\overline{\text{I}}$
1	0	Interrupt request generated at rising edge of $\overline{\text{IRQ7}}$ to $\overline{\text{IF}}$
	1	Interrupt request generated at both falling and rising ed $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ input

5.2.5 IRQ Status Register (ISR)

Bit	7	6	5	4	3	2	1
	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * Only 0 can be written, to clear the flag.

ISR is an 8-bit readable/writable register that indicates the status of IRQ7 to IRQ0 inter requests.

ISR is initialized to H'00 by a reset and in hardware standby mode.

(IRQnSCB = IRQnSCA = 0) and $\overline{\text{IRQn}}$ input is high*

- When IRQn interrupt exception handling is executed while falling, rising, c
  detection is set (IRQnSCB = 1 or IRQnSCA = 1)*

1 [Setting conditions]

- When $\overline{\text{IRQn}}$ input goes low when low-level detection is set (IRQnSCB = IR
- When a falling edge occurs in $\overline{\text{IRQn}}$ input while falling edge detection is se
  (IRQnSCB = 0, IRQnSCA = 1)
- When a rising edge occurs in $\overline{\text{IRQn}}$ input while rising edge detection is set
  (IRQnSCB = 1, IRQnSCA = 0)
- When a falling or rising edge occurs in $\overline{\text{IRQn}}$ input while both-edge detecti
  (IRQnSCB = IRQnSCA = 1)

Notes: n = 7 to 0

* When a product, in which a DTC is incorporated, is used in the following se
corresponding flag bit is not automatically cleared even when exception ha
is a clear 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
automatically cleared.

(2) When DTCEA2 is set to 1 (ICIA is set to an interrupt source), IRQ5F fla
automatically cleared.

(3) When DTCEA1 is set to 1 (ICIB is set to an interrupt source), IRQ6F fla
automatically cleared.

(4) When DTCEA0 is set to 1 (OCIA is set to an interrupt source), IRQ7F fl
automatically cleared.

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

interrupt inputs (pins $\overline{KIN7}$ to $\overline{KIN0}$). To enable key-sense input interrupts from multip
inputs in keyboard matrix scanning/sensing, clear the corresponding mask bits to 0.

KMIMR is initialized to H'BF by a reset and in hardware standby mode and only $\overline{IRQ6}$
input is enabled.

Bits 7 to 0—Keyboard Matrix Interrupt Mask (KMIMR7 to KMIMR0): These bit
key-sense input interrupt requests (KIN7 to KIN0).

Bits 7 to 0 KMIMR7 to KMIMR0	Description
0	Key-sense input interrupt requests enabled
1	Key-sense input interrupt requests disabled (Initial value)*

Note: * However, the initial value of KMIMR6 is 0 because the KMIMR6 bit controls
interrupt request masking and key-sense input enabling.

5.2.7 Keyboard Matrix Interrupt Mask Register (KMIMRA)

Bit	7	6	5	4	3	2	1
	KMIMR15	KMIMR14	KMIMR13	KMIMR12	KMIMR11	KMIMR10	KMIMR9
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

KMIMRA is an 8-bit readable/writable register that performs mask control for the keyb
matrix interrupt inputs (pins $\overline{KIN15}$ to $\overline{KIN8}$). To enable key-sense input interrupts fro
pin inputs in keyboard matrix scanning/sensing, clear the corresponding mask bits to 0.

KMIMRA is initialized to H'FF by a reset and in hardware standby mode.

Figure 5.2 shows the relationship between interrupts IRQ7 and IRQ6, interrupts KIN1
and registers KMIMR and KMIMRA.

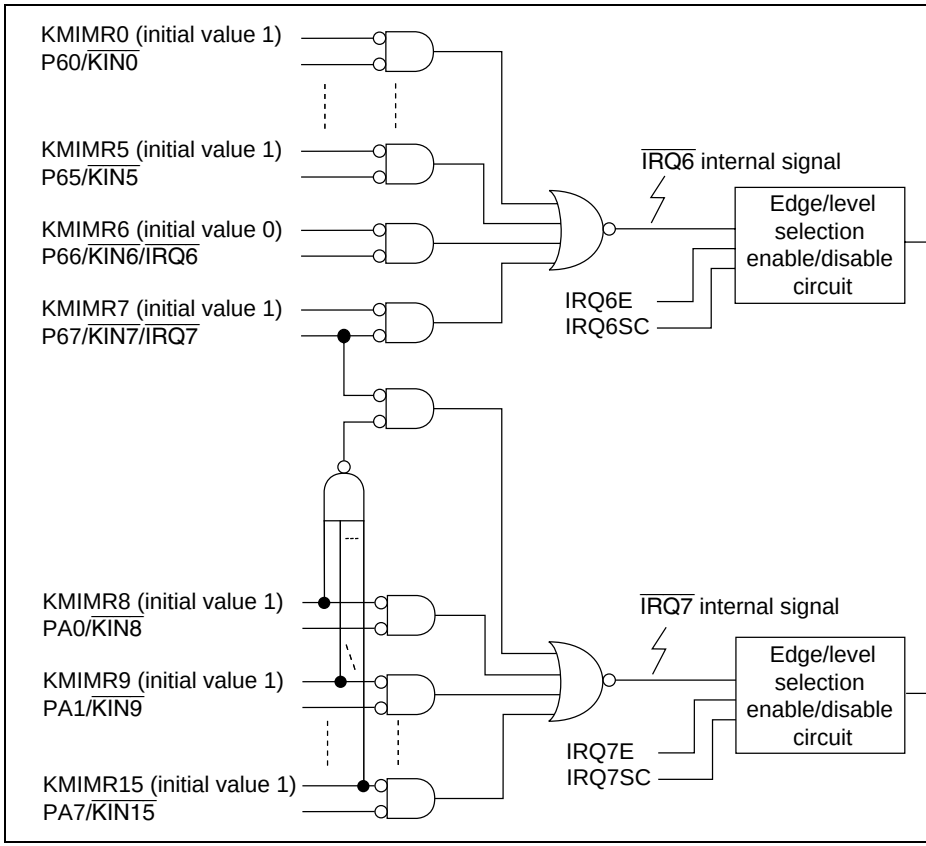


Figure 5.2 Relationship between Interrupts IRQ7 and IRQ6, Interrupts KIN1 and Registers KMIMR and KMIMRA

	CMF	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0
Read/Write	R	—	—	—	—	—	—

ABRKCR is an 8-bit readable/writable register that performs address break control.

ABRKCR is initialized to H'00 by a reset and in hardware standby mode. It is not initia
software standby mode.

Bit 7—Condition Match Flag (CMF): This is the address break source flag, used to i
the address set by BAR has been prefetched. When the CMF flag and BIE flag are both
address break is requested.

Bit 7

CMF	Description
0	[Clearing condition]
	When address break interrupt exception handling is executed
1	[Setting condition]
	When address set by BARA to BARC is prefetched while BIE = 1

Bits 6 to 1—Reserved: These bits cannot be modified and are always read as 0.

Bit 0—Break Interrupt Enable (BIE): Selects address break enabling or disabling.

Bit 0

BIE	Description
0	Address break disabled
1	Address break enabled

Bit	7	6	5	4	3	2	1
BARB	A15	A14	A13	A12	A11	A10	A9
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	7	6	5	4	3	2	1
BARC	A7	A6	A5	A4	A3	A2	A1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BAR consists of three 8-bit readable/writable registers (BARA, BARB, and BARC), a
specify the address at which an address break is to be executed.

Each of the BAR registers is initialized to H'00 by a reset and in hardware standby mo
not initialized in software standby mode.

BARA Bits 7 to 0—Address 23 to 16 (A23 to A16)
BARB Bits 7 to 0—Address 15 to 8 (A15 to A8)
BARC Bits 7 to 1—Address 7 to 1 (A7 to A1)
These bits specify the address at which an address break is to be executed. BAR bits A
are compared with internal address bus lines A23 to A1, respectively.

The address at which the first instruction byte is located should be specified as the bre
Occurrence of the address break condition may not be recognized for other addresses.

In normal mode, no comparison is made with address lines A23 to A16.

BARC Bit 0—Reserved: This bit cannot be modified and is always read as 0.

IRQ6 interrupt source. Of these, NMI, IRQ7, IRQ6, and IRQ2 to IRQ0 can be used to r
H8S/2148 Group or H8S/2144 Group chip from software standby mode.

NMI Interrupt

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

The vector number for NMI interrupt exception handling is 7.

IRQ7 to IRQ0 Interrupts

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

- Using ISCR, it is possible to select whether an interrupt is generated by a low level,
  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.
- The interrupt control level can be set with ICR.
- The status of interrupt requests IRQ7 to IRQ0 is indicated in ISR. ISR flags can be
  by software.

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

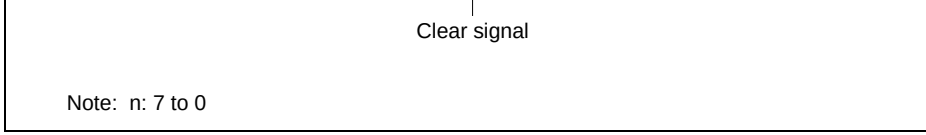


Figure 5.3 Block Diagram of Interrupts IRQ7 to IRQ0

Figure 5.4 shows the timing of IRQnF setting.

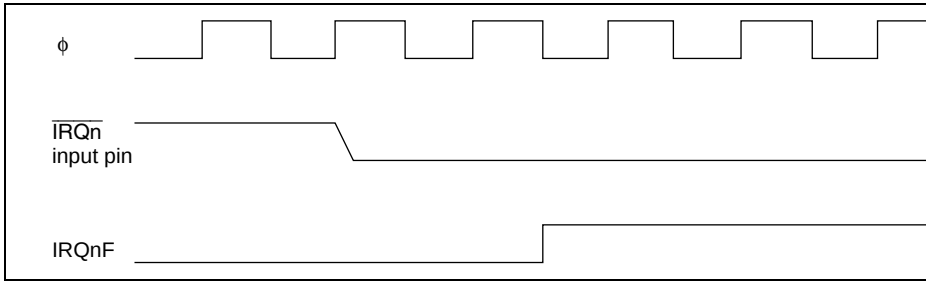


Figure 5.4 Timing of IRQnF Setting

The vector numbers for IRQ7 to IRQ0 interrupt exception handling are 23 to 16.

Detection of IRQ7 to IRQ0 interrupts does not depend on whether the relevant pin has input or output. Therefore, when a pin is used as an external interrupt input pin, clear t corresponding DDR bit to 0 and do not use the pin as an I/O pin for another function. $\overline{\text{IRQ6}}$ pin is assigned as the IRQ6 interrupt input pin, then set the KMIMR6 bit to 0.

When the $\overline{\text{IRQ7}}$ pin is used as the IRQ7 interrupt input pin, bits KMIMR15 to KMIM be set to 1. If any of these bits is cleared to 0, interrupt input from the $\overline{\text{IRQ7}}$ pin will b

As interrupt request flags IRQ7F to IRQ0F are set when the setting condition is met, r the IER setting, only the necessary flags should be referenced.

interrupt request status indications, are all in accordance with the IRQ7 and IRQ6 interr
settings.

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

5.3.2 Internal Interrupts

There are 43 sources for internal interrupts from on-chip supporting modules, plus one
interrupt source (address break).

- For each on-chip supporting module there are flags that indicate the interrupt reques
  and enable bits that select enabling or disabling of these interrupts. If any one of the
  1, an interrupt request is issued to the interrupt controller.
- The interrupt control level can be set by means of ICR.
- The DTC can be activated by an FRT, TMR, SCI, or other interrupt request. When
  activated by an interrupt, the interrupt control mode and interrupt mask bits have no

5.3.3 Interrupt Exception Vector Table

Table 5.4 shows interrupt exception handling sources, vector addresses, and interrupt p
For default priorities, the lower the vector number, the higher the priority.

Priorities among modules can be set by means of ICR. The situation when two or more
are set to the same priority, and priorities within a module, are fixed as shown in table 5

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		23	H'002E	H'00005C	
SWDTEND (software activation interrupt end)	DTC	24	H'0030	H'000060	ICRA
WOVI0 (interval timer)	Watchdog timer 0	25	H'0032	H'000064	ICRA
WOVI1 (interval timer)	Watchdog timer 1	26	H'0034	H'000068	ICRA
Address break (PC break)	—	27	H'0036	H'00006C	
ADI (A/D conversion end)	A/D	28	H'0038	H'000070	ICRE
Reserved	—	29 to 47	H'003A to H'005E	H'000074 to H'0000BC	
ICIA (input capture A)	Free-running timer	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		55	H'006E	H'0000DC	
Reserved	—	56 to 63	H'0070 to H'007E	H'0000E0 to H'0000FC	

CMIA1 (compare-match A)	8-bit timer channel 1	68	H'0088	H'000110	ICRB
CMIB1 (compare-match B)		69	H'008A	H'000114	
OVI1 (overflow)		70	H'008C	H'000118	
Reserved		71	H'008E	H'00011C	
CMIAY (compare-match A)	8-bit timer channels Y, X	72	H'0090	H'000120	ICRB
CMIBY (compare-match B)		73	H'0092	H'000124	
OVIY (overflow)		74	H'0094	H'000128	
ICIX (input capture X)		75	H'0096	H'00012C	
IBF1 (IDR1 reception completed)	Host interface	76	H'0098	H'000130	ICRB
IBF2 (IDR2 reception completed)		77	H'009A	H'000134	
IBF3 (IDR3 reception completed)		78	H'009C	H'000138	
IBF4 (IDR4 reception completed)		79	H'009E	H'00013C	
ERI0 (receive error 0)	SCI channel 0	80	H'00A0	H'000140	ICRC
RXI0 (reception completed 0)		81	H'00A2	H'000144	
TXI0 (transmit data empty 0)		82	H'00A4	H'000148	
TEI0 (transmission end 0)		83	H'00A6	H'00014C	
ERI1 (receive error 1)	SCI channel 1	84	H'00A8	H'000150	ICRC
RXI1 (reception completed 1)		85	H'00AA	H'000154	
TXI1 (transmit data empty 1)		86	H'00AC	H'000158	
TEI1 (transmission end 1)		87	H'00AE	H'00015C	
ERI2 (receive error 2)	SCI channel 2	88	H'00B0	H'000160	ICRC
RXI2 (reception completed 2)		89	H'00B2	H'000164	
TXI2 (transmit data empty 2)		90	H'00B4	H'000168	
TEI2 (transmission end 2)		91	H'00B6	H'00016C	
IICI0 (1-byte transmission/ reception completed)	IIC channel 0 (option)	92	H'00B8	H'000170	ICRC
DDCSWI (format switch)		93	H'00BA	H'000174	

PS2IC (reception completed C)	controller (PS2)	98	H'00C4	H'000188
Reserved		99	H'00C6	H'00018C
Reserved	—	100 to 103	H'00C8 to H'00CE	H'000190 to H'00019C

This function can be used to detect the beginning of execution of a bug location in the p
and branch to a correction routine.

5.4.2 Block Diagram

A block diagram of the address break function is shown in figure 5.5.

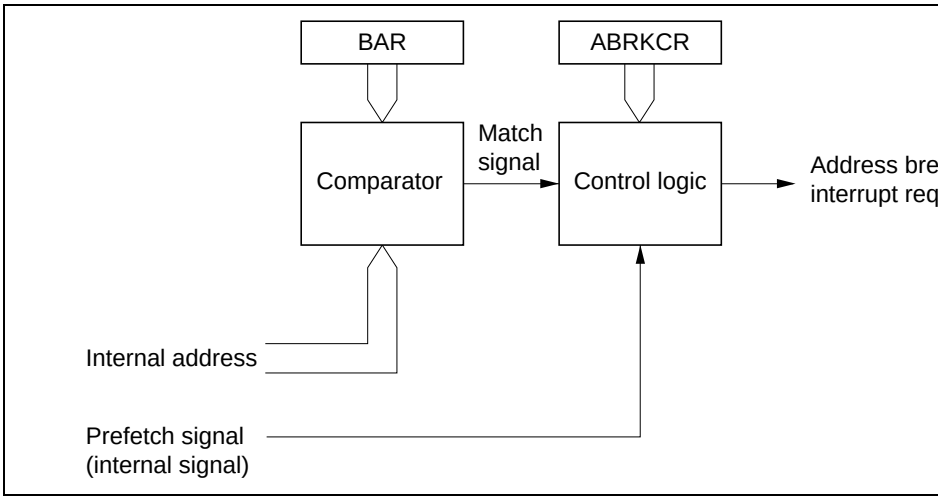


Figure 5.5 Block Diagram of Address Break Function

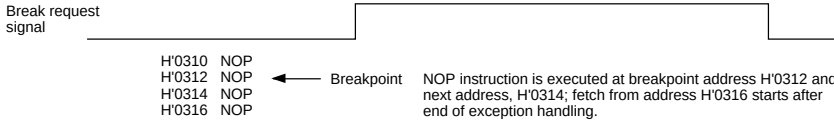
The register settings when the address break function is used are as follows.

1. Set the break address in bits A23 to A1 in BAR.
2. Set the BIE bit in ABRKCR to 1 to enable address breaks. An address break will n
   requested if the BIE bit is cleared to 0.

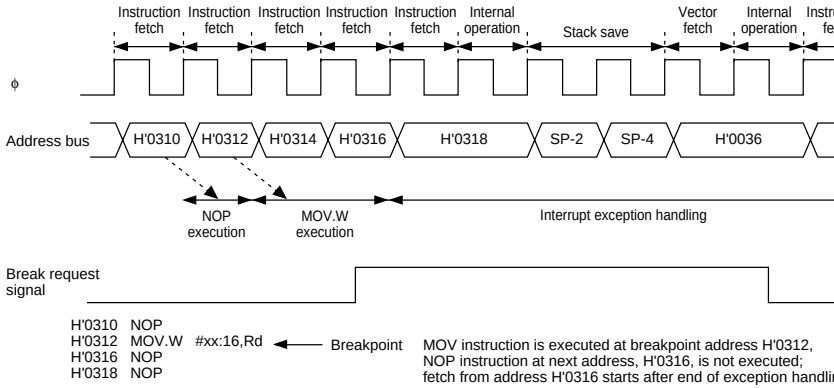
When the setting condition occurs, the CMF flag in ABRKCR is set to 1 and an interr
requested. If necessary, the source should be identified in the interrupt handling routin

5.4.4 Usage Notes

- With the address break function, the address at which the first instruction byte is lo
  should be specified as the break address. Occurrence of the address break conditio
  recognized for other addresses.
- In normal mode, no comparison is made with address lines A23 to A16.
- If a branch instruction (Bcc, BSR), jump instruction (JMP, JSR), RTS instruction,
  instruction is located immediately before the address set in BAR, execution of this
  will output a prefetch signal for that address, and an address break may be request
  be prevented by not making a break address setting for an address immediately fol
  of these instructions, or by determining within the interrupt handling routine whetl
  handling was initiated by a genuine condition occurrence.
- As an address break interrupt is generated by a combination of the internal prefetc
  address, the timing of the start of interrupt exception handling depends on the cont
  execution cycle of the instruction at the set address and the preceding instruction. I
  shows some address timing examples.



- Program area in on-chip memory, 2-state execution instruction at specified break address



- Program area in external memory (2-state access, 16-bit-bus access), 1-state execution instruction at specified break address

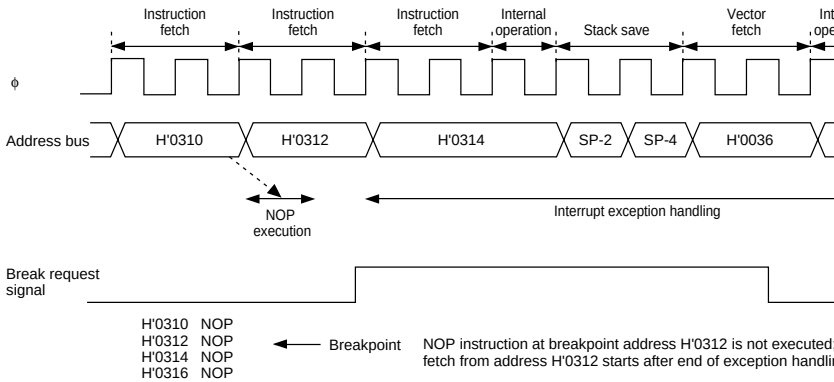


Figure 5.6 Examples of Address Break Timing

enable bit is provided for each interrupt. Clearing an enable bit to 0 disables the corres
interrupt request. Interrupt sources for which the enable bits are set to 1 are controlled
interrupt controller.

Table 5.5 shows the interrupt control modes.

The interrupt controller performs interrupt control according to the interrupt control m
the INTM1 and INTM0 bits in SYSCR, the priorities set in ICR, and the masking stat
by the I and UI bits in the CPU's CCR.

Table 5.5 Interrupt Control Modes

Interrupt Control Mode	SYSCR		Priority Setting Register	Interrupt Mask Bits	Description
	INTM1	INTM0			
0	0	0	ICR	I	Interrupt mask cor performed by the I Priority can be set
1		1	ICR	I, UI	3-level interrupt m is performed by th bits Priority can be set

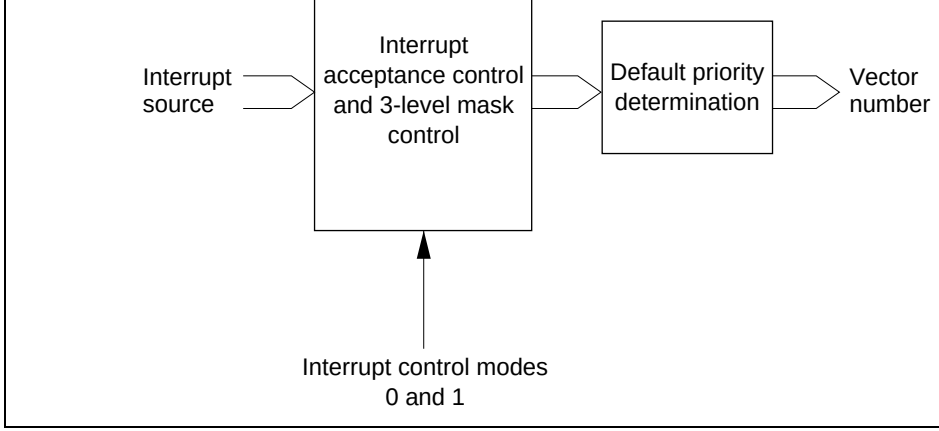


Figure 5.7 Block Diagram of Interrupt Control Operation

Interrupt Acceptance Control and 3-Level Control

In interrupt control modes 0 and 1, interrupt acceptance control and 3-level mask contr performed by means of the I and UI bits in CCR, and ICR (control level).

Table 5.6 shows the interrupts selected in each interrupt control mode.

		interrupts
	1	NMI, and address break interru

Legend:
*: Don’t care

Default Priority Determination

The priority is determined for the selected interrupt, and a vector number is generated

If the same value is set for ICR, acceptance of multiple interrupts is enabled, and so o
interrupt source with the highest priority according to the preset default priorities is se
has a vector number generated.

Interrupt sources with a lower priority than the accepted interrupt source are held pend

Table 5.7 shows operations and control signal functions in each interrupt control mod

Table 5.7 Operations and Control Signal Functions in Each Interrupt Control

Interrupt Control Mode	Setting		Interrupt Acceptance Control 3-Level Control				Default Priority Determination
	INTM1	INTM0		I	UI	ICR	
0	0	0	O	IM	—	PR	O
1	0	1	O	IM	IM	PR	O

Legend:
O: Interrupt operation control performed
IM: Used as interrupt mask bit
PR: Sets priority
—: Not used

interrupt request is sent to the interrupt controller.

2. When interrupt requests are sent to the interrupt controller, a control level 1 interrup according to the control level set in ICR, has priority for selection, and other interru are held pending. If a number of interrupt requests with the same control level settin generated at the same time, the interrupt request with the highest priority according priority system shown in table 5.4 is selected.

3. The I bit is then referenced. If the I bit is cleared to 0, the interrupt request is accept bit is set to 1, only NMI and address break interrupt are accepted, and other interrup are held pending.

4. When an interrupt request is accepted, interrupt exception handling starts after exec current instruction has been completed.

5. The PC and CCR are saved to the stack area by interrupt exception handling. The P the stack shows the address of the first instruction to be executed after returning fro interrupt handling routine.

6. Next, the I bit in CCR is set to 1. This disables all interrupts except NMI and addres

7. A vector address is generated for the accepted interrupt, and execution of the interru handling routine starts at the address indicated by the contents of that vector addres

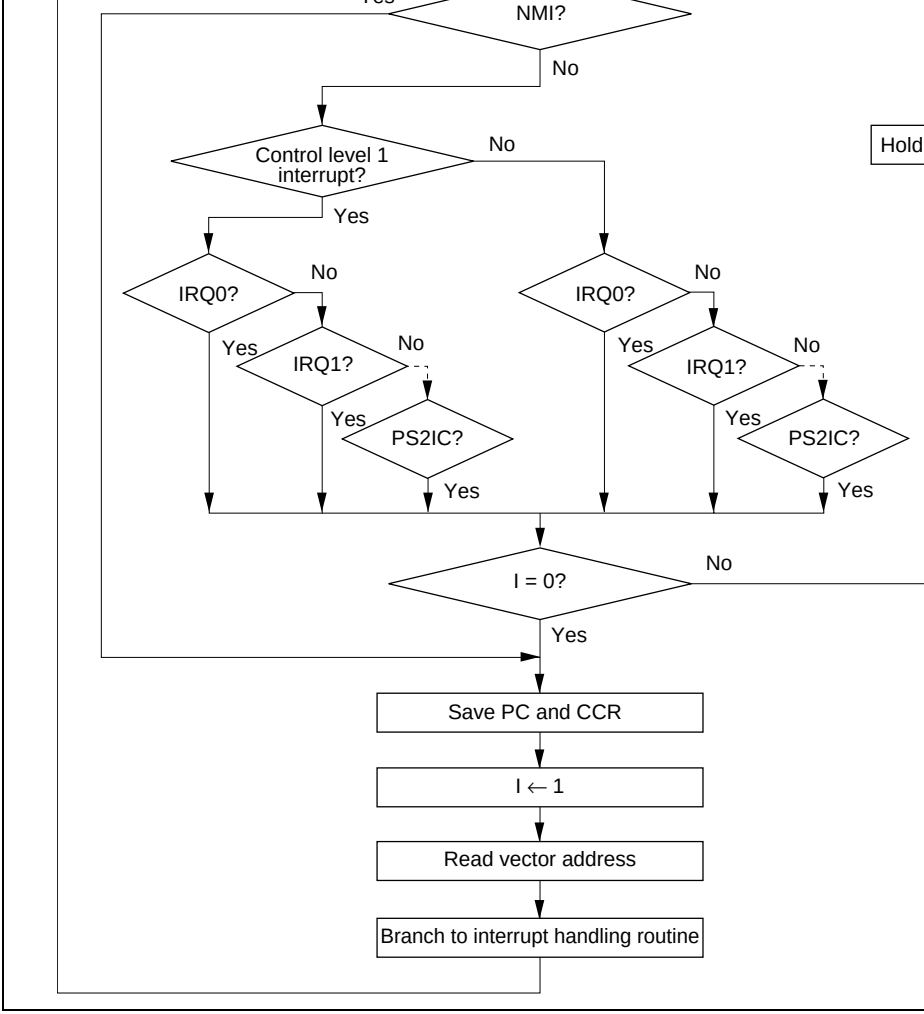


Figure 5.8 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0

disabled when both the I bit and the UI bit are set to 1.

For example, if the interrupt enable bit for an interrupt request is set to 1, and H'20, H'0 are set in ICRA, ICRB, and ICRC, respectively, (i.e. IRQ2 and IRQ3 interrupts are set level 1 and other interrupts to control level 0), the situation is as follows:

- When I = 0, all interrupts are enabled
  (Priority order: NMI > IRQ2 > IRQ3 > address break > IRQ0 > IRQ1 ...)
- When I = 1 and UI = 0, only NMI, IRQ2, IRQ3 and address break interrupts are ena
- When I = 1 and UI = 1, only NMI and address break interrupts are enabled

Figure 5.9 shows the state transitions in these cases.

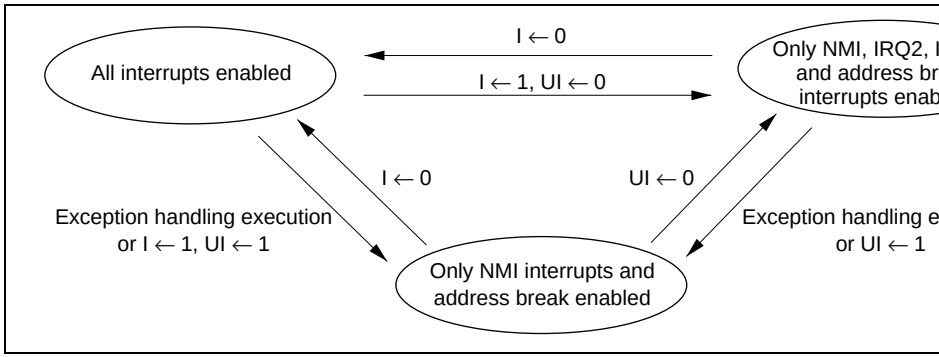


Figure 5.9 Example of State Transitions in Interrupt Control Mode 1

priority system shown in table 5.4 is selected.

3. The I bit is then referenced. If the I bit is cleared to 0, the UI bit has no effect.

   An interrupt request set to interrupt control level 0 is accepted when the I bit is cle
   the I bit is set to 1, only an NMI and address break interrupts are accepted, and oth
   requests are held pending.

   An interrupt request set to interrupt control level 1 has priority over an interrupt re
   interrupt control level 0, and is accepted if the I bit is cleared to 0, or if the I bit is
   the UI bit is cleared to 0.

   When both the I bit and the UI bit are set to 1, only an NMI and address break inte
   accepted, and other interrupt requests are held pending.

4. When an interrupt request is accepted, interrupt exception handling starts after exe
   current instruction has been completed.

5. The PC and CCR are saved to the stack area by interrupt exception handling. The
   the stack shows the address of the first instruction to be executed after returning fr
   interrupt handling routine.

6. Next, the I and UI bits in CCR are set to 1. This disables all interrupts except NMI
   break.

7. A vector address is generated for the accepted interrupt, and execution of the inter
   handling routine starts at the address indicated by the contents of that vector addre

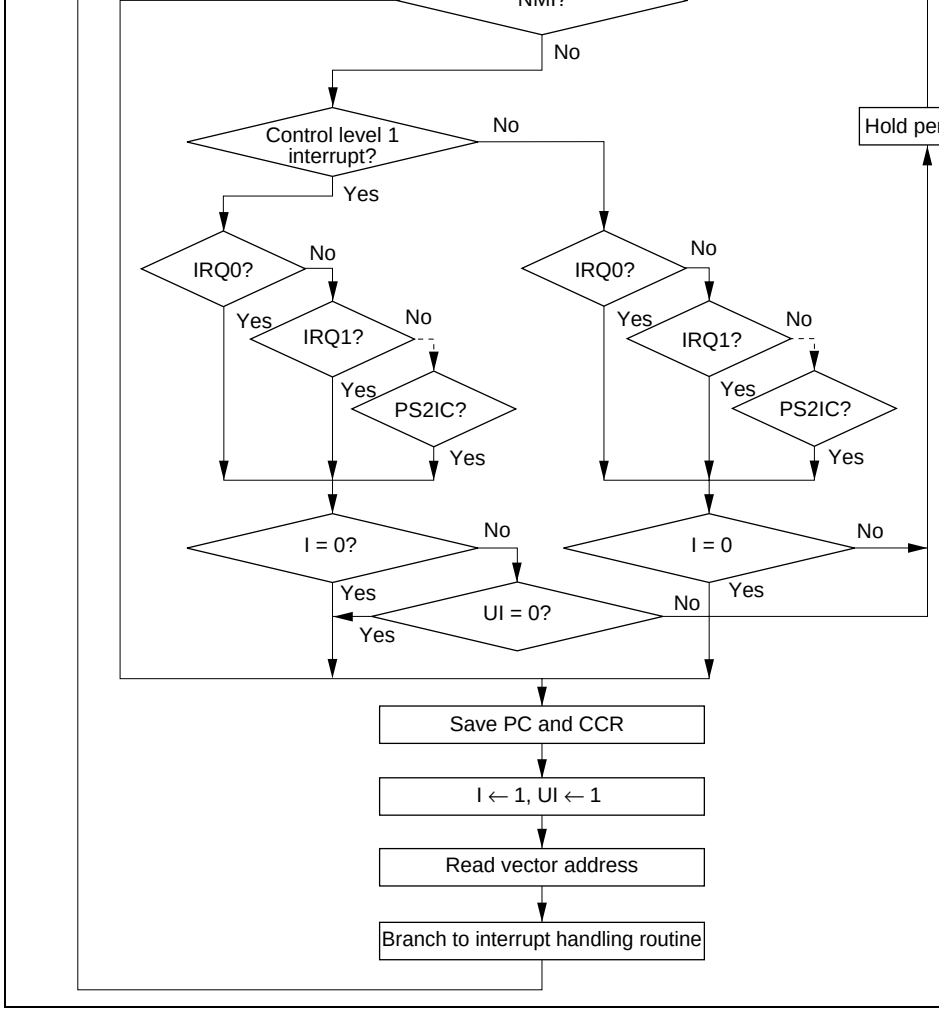


Figure 5.10 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 1

Figure 5.11 Interrupt Exception Handling

Table 5.8 Interrupt Response Times

No.	Item	Number of States: Normal Mode	Number of States: Advanced
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 interrupt handling routine prefetch.
4. Internal processing after interrupt acceptance and internal processing after

Table 5.9 Number of States in Interrupt Handling Routine Execution

		Object of Access			
			External Device		
			8-Bit Bus		16-B
	Symbol	Internal Memory	2-State Access	3-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 an external device access

MOV, if an interrupt is generated during execution of the instruction, the interrupt cond
still be enabled on completion of the instruction, and so interrupt exception handling fo
interrupt will be executed on completion of the instruction. However, if there is an inte
request of higher priority than that interrupt, interrupt exception handling will be execu
higher-priority interrupt, and the lower-priority interrupt will be ignored.

The same also applies when an interrupt source flag is cleared to 0.

Figure 5.12 shows an example in which the CMIEA bit in 8-bit timer register TCR is cl

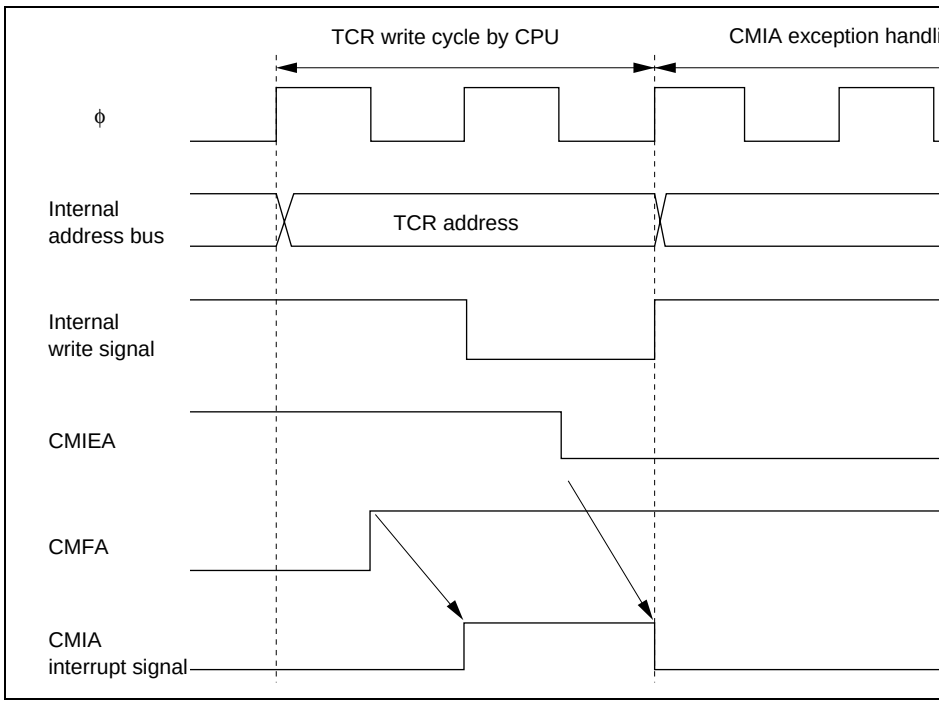


Figure 5.12 Contention between Interrupt Generation and Disabling

two states after execution of the instruction ends.

5.6.3 Interrupts during Execution of EEPMOV Instruction

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

With the EEPMOV.B instruction, an interrupt request (including NMI) issued during 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 st case is the address of the next instruction.

Therefore, if an interrupt is generated during execution of an EEPMOV.W instruction following coding should be used.

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

- Both of the above

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

5.7.2 Block Diagram

Figure 5.13 shows a block diagram of the DTC and interrupt controller.

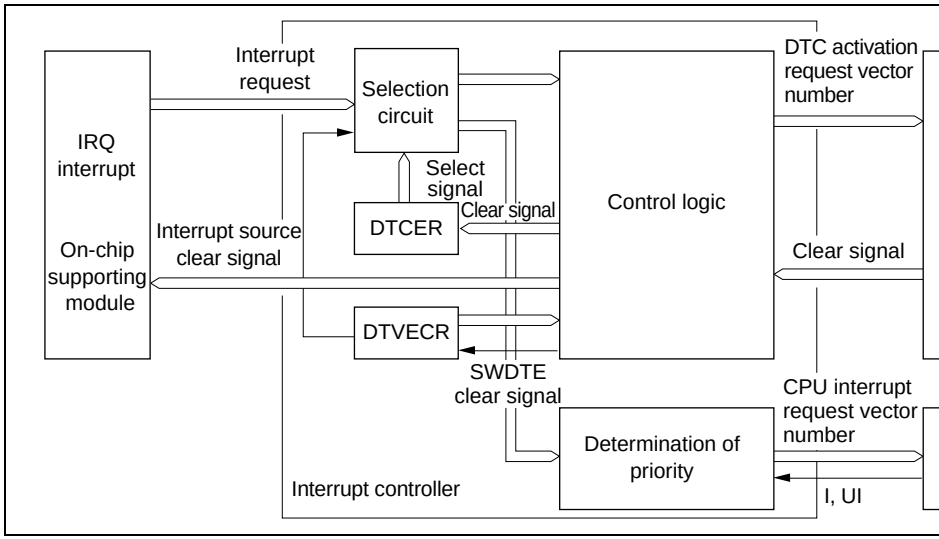


Figure 5.13 Interrupt Control for DTC

following the DTC data transfer the DTCE bit is cleared to 0 and an interrupt request
CPU.

Determination of Priority: The DTC activation source is selected in accordance with
priority order, and is not affected by mask or priority levels. See section 7.3.3, DTC V
for the respective priorities.

Operation Order: If the same interrupt is selected as a DTC activation source and a
interrupt source, the DTC data transfer is performed first, followed by CPU interrupt e
handling.

Table 5.10 summarizes interrupt source selection and interrupt source clearance contr
to the settings of the DTCE bit of DTCERA to DTCERE in the DTC and the DISEL b
in the DTC.

Table 5.10 Interrupt Source Selection and Clearing Control

Settings			
DTC		Interrupt Source Selection/Clearin	
DTCE	DISEL	DTC	CPU
0	*	×	Δ
1	0	Δ	×
	1	○	Δ

Legend:

Δ: 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 bit cannot be used.

*: Don't care

The bus controller also has a bus arbitration function, and controls the operation of the masters: the CPU and data transfer controller (DTC).

6.1.1 Features

The features of the bus controller are listed below.

- Basic bus interface
  - 2-state access or 3-state access can be selected
  - Program wait states can be inserted
- Burst ROM interface
  - External space can be designated as ROM interface space
  - 1-state or 2-state burst access can be selected
- Idle cycle insertion
  - An idle cycle can be inserted when an external write cycle immediately follow read cycle
- Bus arbitration function
  - Includes a bus arbiter that arbitrates bus mastership between the CPU and DTC

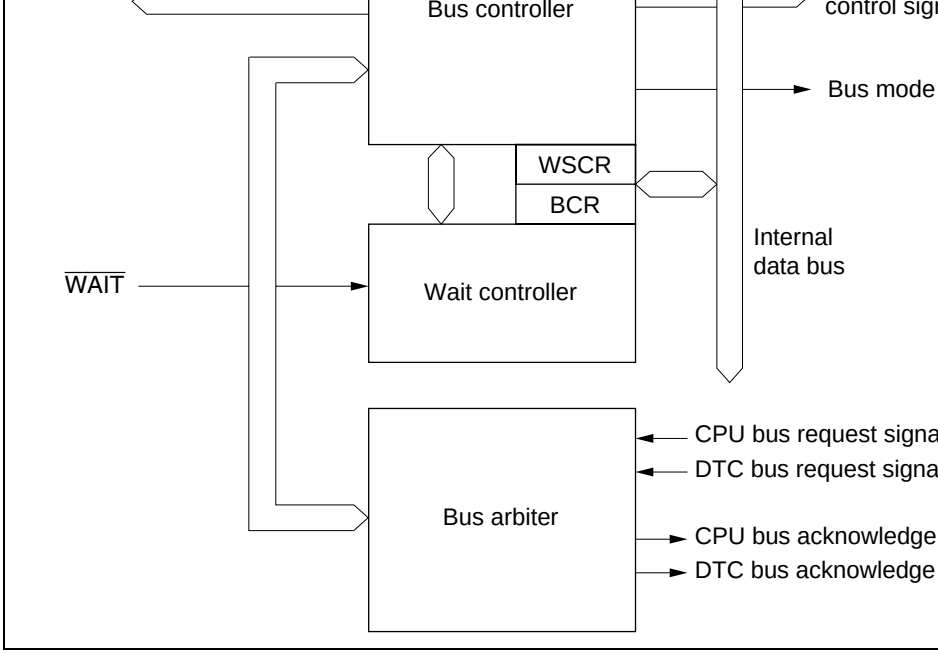


Figure 6.1 Block Diagram of Bus Controller

I/O select	$\overline{\text{IOS}}$	Output	I/O select signal (when IOSE bit is
Read	$\overline{\text{RD}}$	Output	Strobe signal indicating that extern being read
High write	$\overline{\text{HWR}}$	Output	Strobe signal indicating that extern being written to, and that the uppe (D15 to D8) is enabled
Low write	$\overline{\text{LWR}}$	Output	Strobe signal indicating that extern being written to, and that the lowe (D7 to D0) is enabled
Wait	$\overline{\text{WAIT}}$	Input	Wait request signal when external access space is accessed

6.1.4 Register Configuration

Table 6.2 summarizes the registers of the bus controller.

Table 6.2 Bus Controller Registers

Name	Abbreviation	R/W	Initial Value	Ad
Bus control register	BCR	R/W	H'D7	H'F
Wait state control register	WSCR	R/W	H'33	H'F

Note: * Lower 16 bits of the address.

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BCR is an 8-bit readable/writable register that specifies the external memory space acc
and the extent of the I/O area when the I/O strobe function has been selected for the $\overline{AS}$

BCR is initialized to H'D7 by a reset and in hardware standby mode. It is not initialized
software standby mode.

Bit 7—Idle Cycle Insert 1 (ICIS1): Reserved. Do not write 0 to this bit.

Bit 6—Idle Cycle Insert 0 (ICIS0): Selects whether or not a one-state idle cycle is to
between bus cycles when successive external read and external write cycles are perform

Bit 6

ICIS0	Description
0	Idle cycle not inserted in case of successive external read and external writ
1	Idle cycle inserted in case of successive external read and external write cy (I

Bit 5—Burst ROM Enable (BRSTRM): Selects whether external space is designated
ROM interface space. The selection applies to the entire external space.

Bit 5

BRSTRM	Description
0	Basic bus interface (I
1	Burst ROM interface

Bit 3—Burst Cycle Select 0 (BRSTS0): Selects the number of words that can be acc
burst ROM interface burst access.

Bit 3

BRSTS0	Description
0	Max. 4 words in burst access
1	Max. 8 words in burst access

Bit 2—Reserved: Do not write 0 to this bit.

Bits 1 and 0—IOS Select 1 and 0 (IOS1, IOS0): See table 6.4.

6.2.2 Wait State Control Register (WSCR)

Bit	7	6	5	4	3	2	1
	RAMS	RAM0	ABW	AST	WMS1	WMS0	WC1
Initial value	0	0	1	1	0	0	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

WSCR is an 8-bit readable/writable register that specifies the data bus width, number
states, wait mode, and number of wait states for external memory space. The on-chip
internal I/O register bus width and number of access states are fixed, irrespective of th
settings.

WSCR is initialized to H'33 by a reset and in hardware standby mode. It is not initializ
software standby mode.

0	External memory space is designated as 16-bit access space	
1	External memory space is designated as 8-bit access space	(In

Bit 4—Access State Control (AST): Specifies whether the external memory space is 2 access space or 3-state access space, and simultaneously enables or disables wait state i

Bit 4

AST	Description	
0	External memory space is designated as 2-state access space Wait state insertion in external memory space accesses is disabled	
1	External memory space is designated as 3-state access space Wait state insertion in external memory space accesses is enabled	(In

Bits 3 and 2—Wait Mode Select 1 and 0 (WMS1, WMS0): These bits select the wait when external memory space is accessed while the AST bit is set to 1.

Bit 3	Bit 2		
WMS1	**WMS0**	**Description**	
0	0	Program wait mode	(In
	1	Wait-disabled mode	
1	0	Pin wait mode	
	1	Pin auto-wait mode	

1	3 program wait states are inserted in external memory space

6.3 Overview of Bus Control

6.3.1 Bus Specifications

The external space bus specifications consist of three elements: bus width, number of states, and wait mode and number of program wait states.

The bus width and number of access states for on-chip memory and internal I/O regist fixed, and are not affected by the bus controller.

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

Number of Access States: Two or three access states can be selected with the AST bi When 2-state access space is designated, wait insertion is disabled. The number of acc the burst ROM interface is determined without regard to the AST bit setting.

Wait Mode and Number of Program Wait States: When 3-state access space is des the AST bit, the wait mode and the number of program wait states to be inserted autor selected with WMS1, WMS0, WC1, and WC0. From 0 to 3 program wait states can b

Table 6.3 shows the bus specifications for each basic bus interface area.

					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: * Except when WMS1 = 0 and WMS0 = 1

6.3.2 Advanced Mode

The initial state of the external space is basic bus interface, three-state access space. In enabled expanded mode, the space excluding the on-chip ROM, on-chip RAM, and inte registers is external space. The on-chip RAM is enabled when the RAME bit in the syst register (SYSCR) is set to 1; when the RAME bit is cleared to 0, the on-chip RAM is d the corresponding space becomes external space.

6.3.3 Normal Mode

The initial state of the external memory space is basic bus interface, three-state access s ROM-disabled expanded mode, the space excluding the on-chip RAM and internal I/O external space. In ROM-enabled expanded mode, the space excluding the on-chip ROM RAM, and internal I/O registers is external space. The on-chip RAM is enabled when th bit in the system control register (SYSCR) is set to 1; when the RAME bit is cleared to chip RAM is disabled and the corresponding space becomes external space.

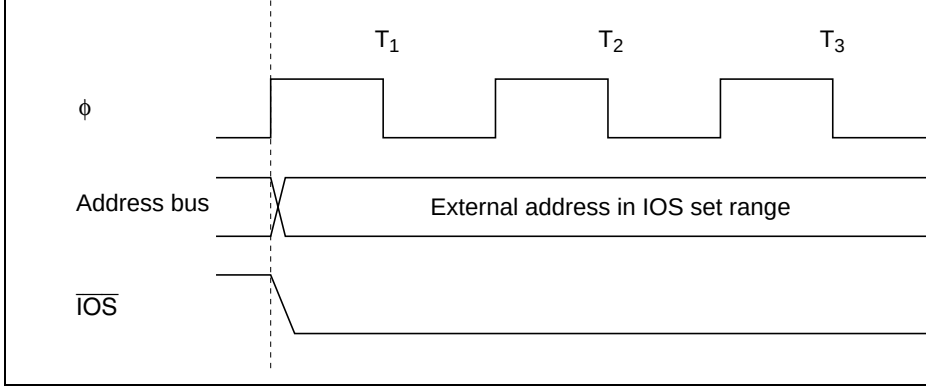


Figure 6.2 $\overline{\text{IOS}}$ Signal Output Timing

Enabling or disabling of $\overline{\text{IOS}}$ signal output is controlled by the setting of the IOSE bit
In expanded mode, this pin operates as the $\overline{\text{AS}}$ output pin after a reset, and therefore th
in SYSCR must be set to 1 in order to use this pin as the $\overline{\text{IOS}}$ signal output. See sectio
Ports, for details.

The range of addresses for which the $\overline{\text{IOS}}$ signal is output can be set with bits IOS1 an
BCR. The $\overline{\text{IOS}}$ signal address ranges are shown in table 6.4.

Table 6.4 $\overline{\text{IOS}}$ Signal Output Range Settings

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)FE4F*	(In

Note: * In the H8S/2148 and H8S/2147 F-ZTAT A-mask version, the address range
H'(FF)F000 to H'(FF)F7FF.

6.4.2 Data Size and Data Alignment

Data sizes for the CPU and other internal bus masters are byte, word, and longword. Th
controller has a data alignment function, and when accessing external space, controls w
upper data bus (D15 to D8) or lower data bus (D7 to D0) is used according to the bus
specifications for the area being accessed (8-bit access space or 16-bit access space) an
size.

8-Bit Access Space

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

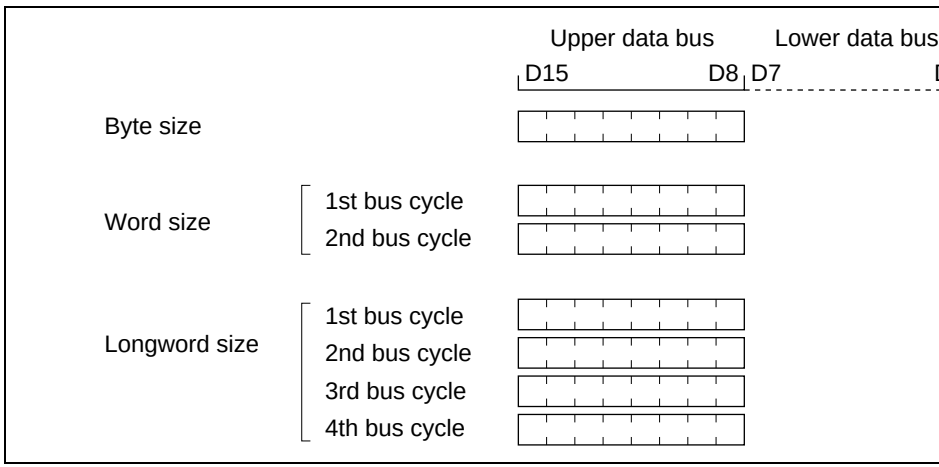


Figure 6.3 Access Sizes and Data Alignment Control (8-Bit Access Spac

address.

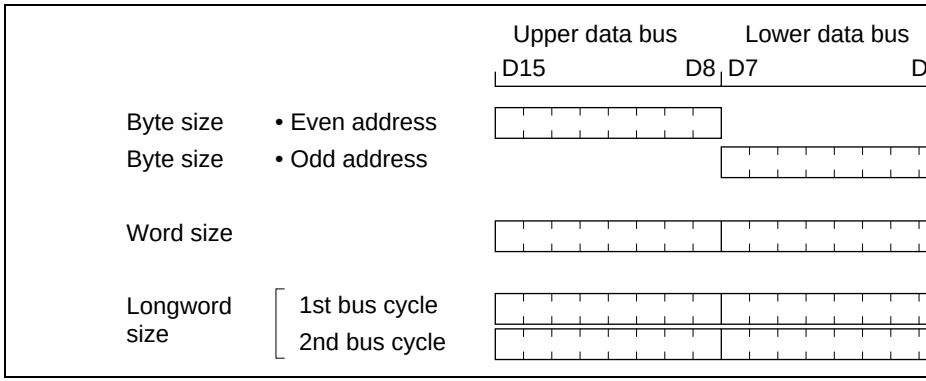


Figure 6.4 Access Sizes and Data Alignment Control (16-Bit Access Spa

Table 6.5 Data Buses Used and Valid Strobes

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

Legend:

Undefined: Undefined data is output.

Invalid: Input state; input value is ignored.

Port, etc.: Pins are used as port or on-chip supporting module input/output pins, and n bus pins.

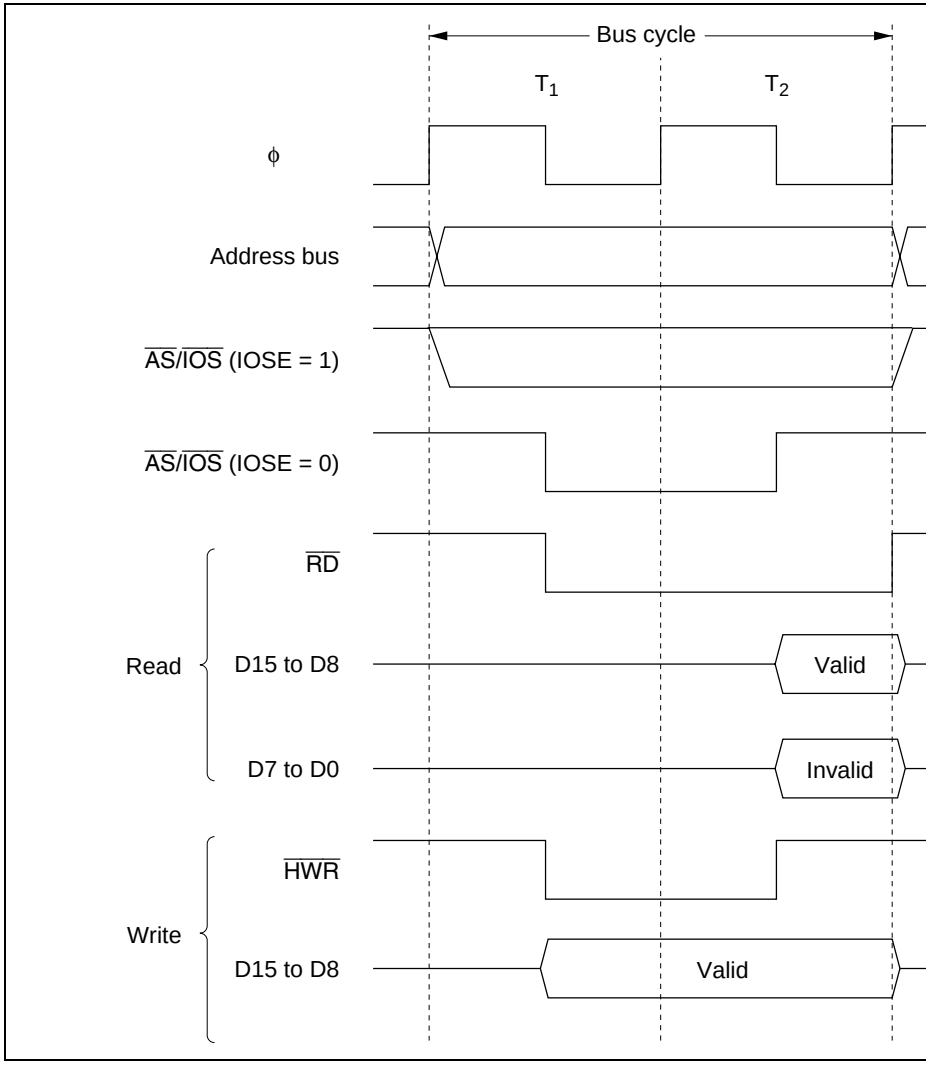


Figure 6.5 Bus Timing for 8-Bit 2-State Access Space

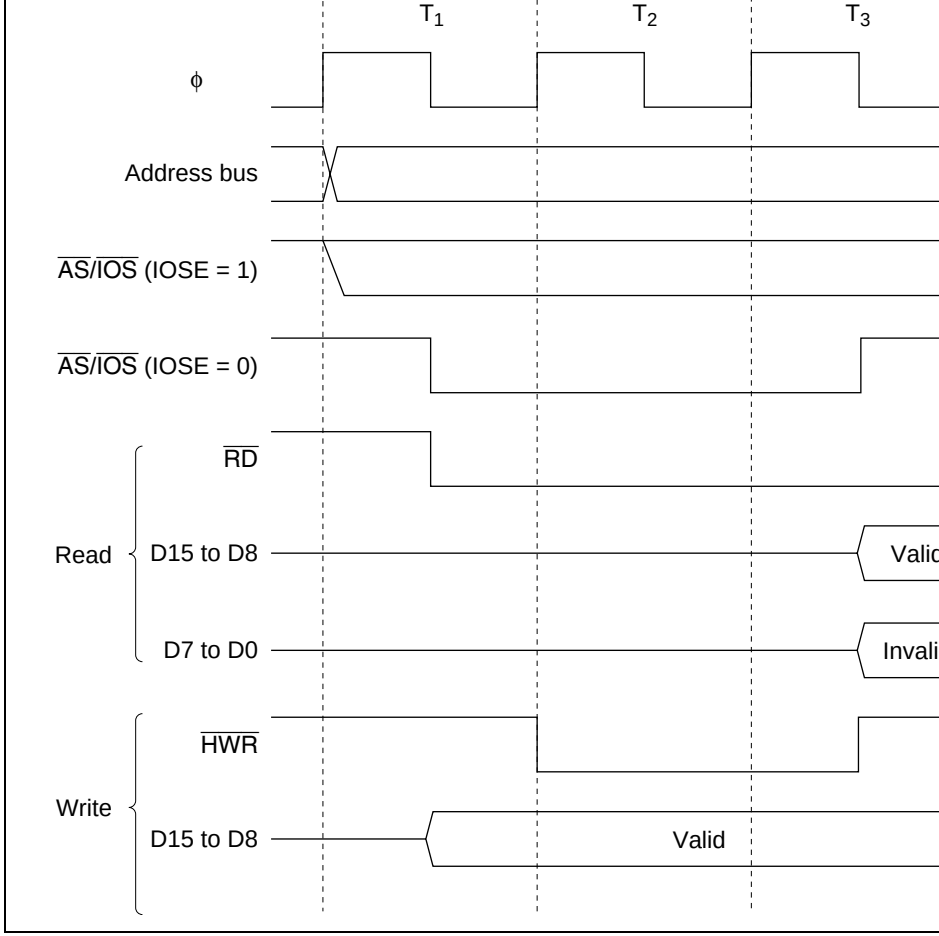


Figure 6.6 Bus Timing for 8-Bit 3-State Access Space

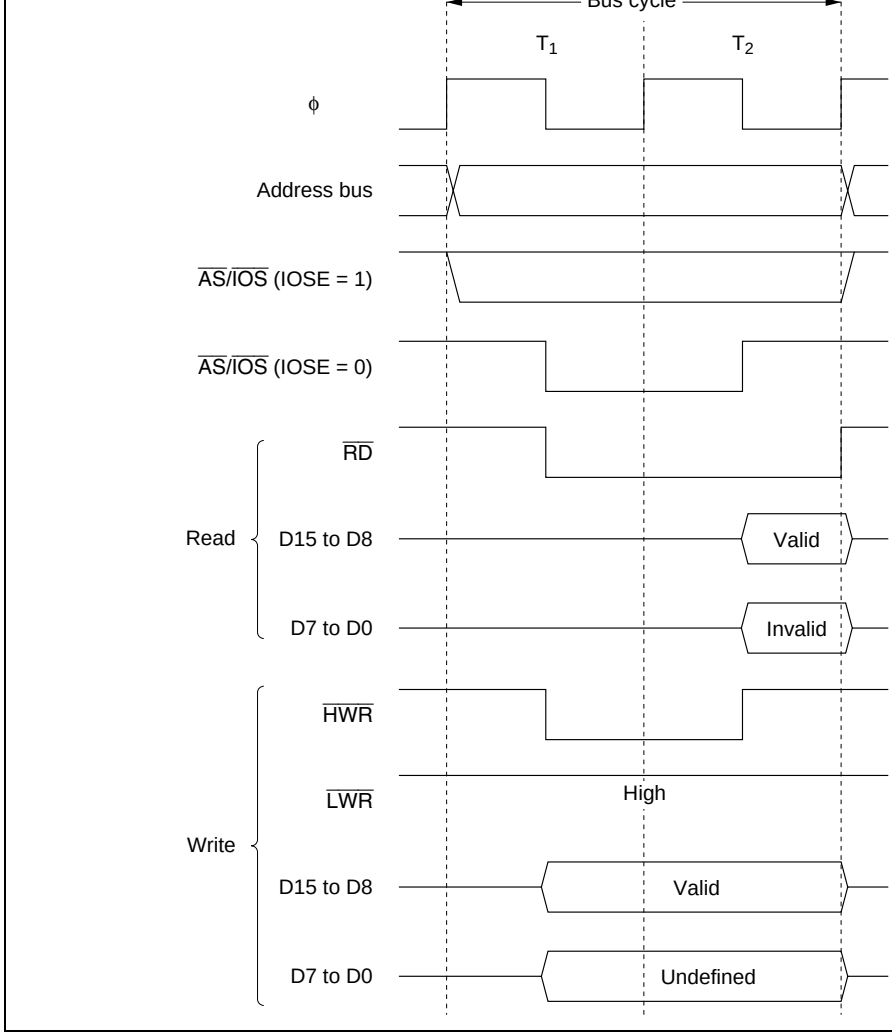


Figure 6.7 16-Bit, 2-State Access Space Bus Timing (1)
(Even Address Byte Access)

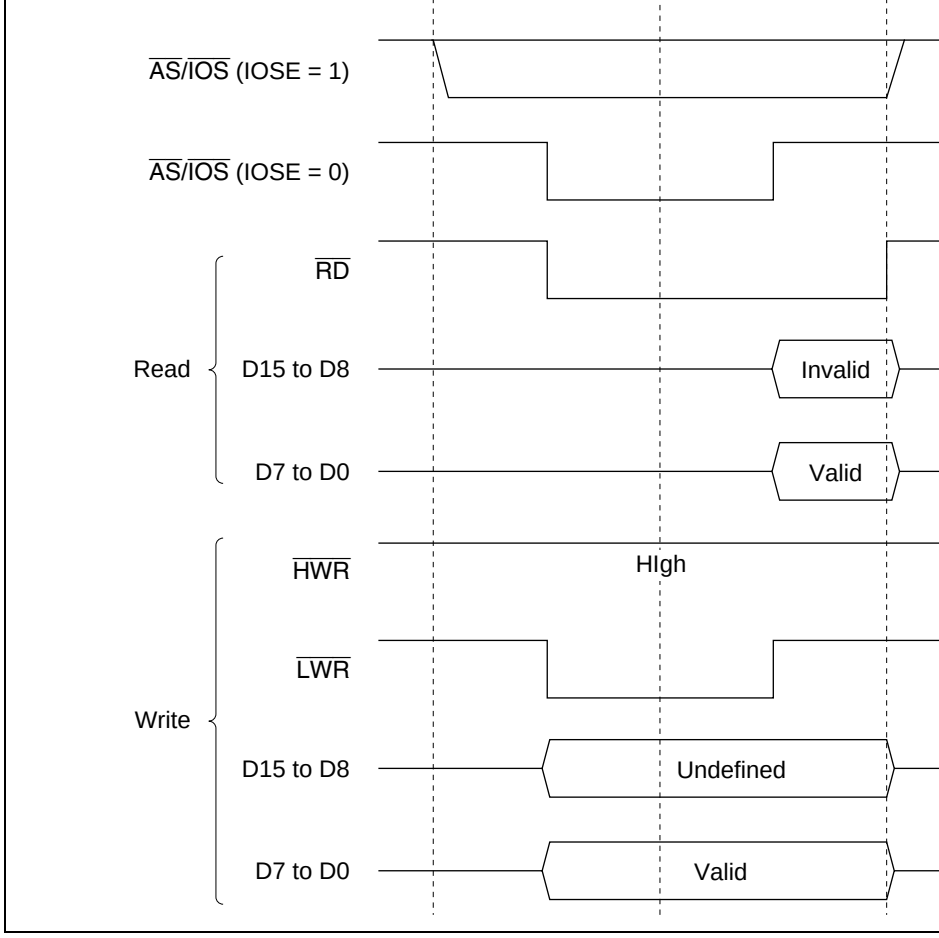


Figure 6.8 16-Bit, 2-State Access Space Bus Timing (2)
(Odd Address Byte Access)

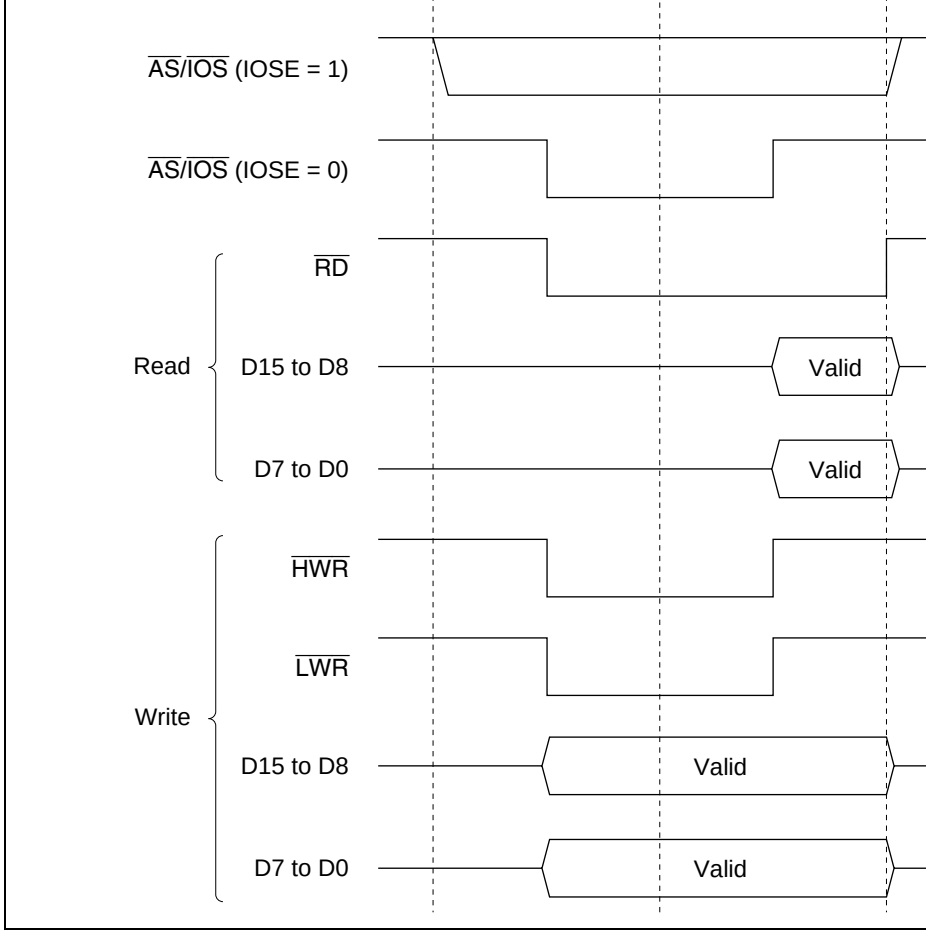


Figure 6.9 16-Bit, 2-State Access Space Bus Timing (3) (Word Access)

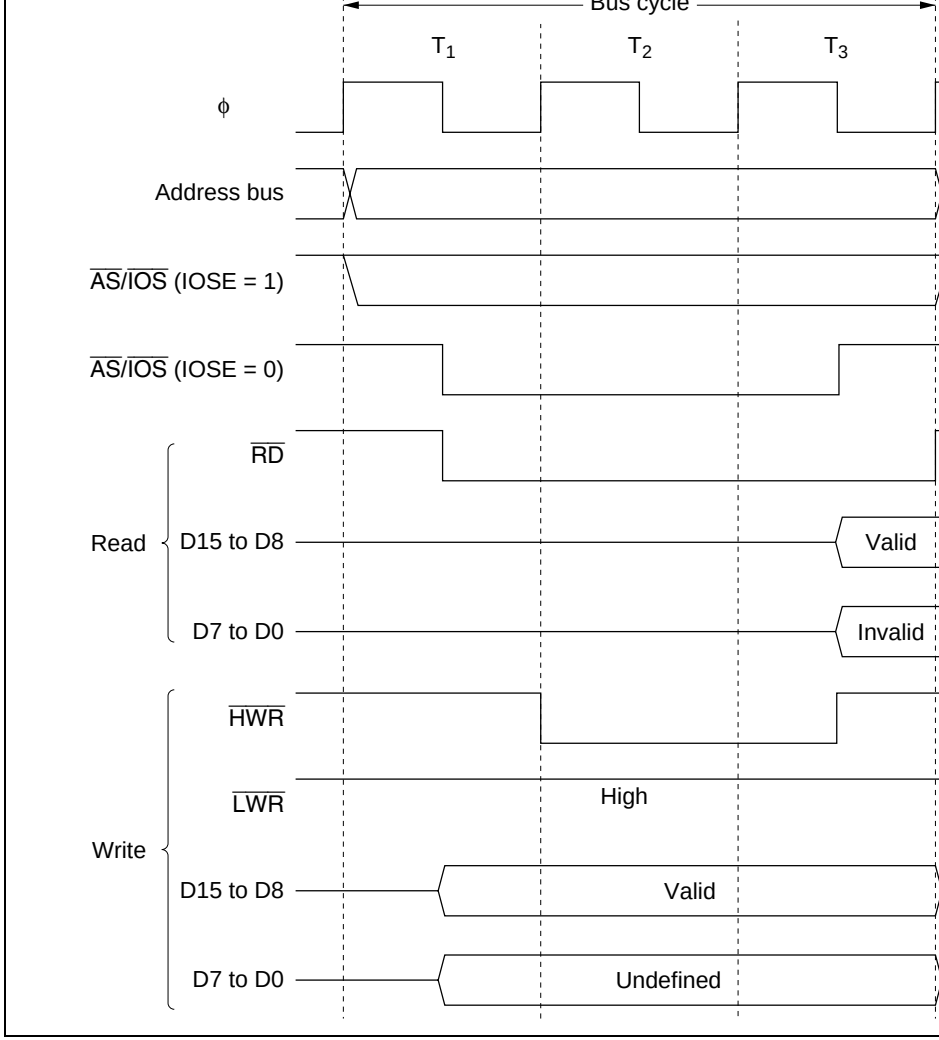


Figure 6.10 16-Bit, 3-State Access Space Bus Timing (1)
(Even Address Byte Access)

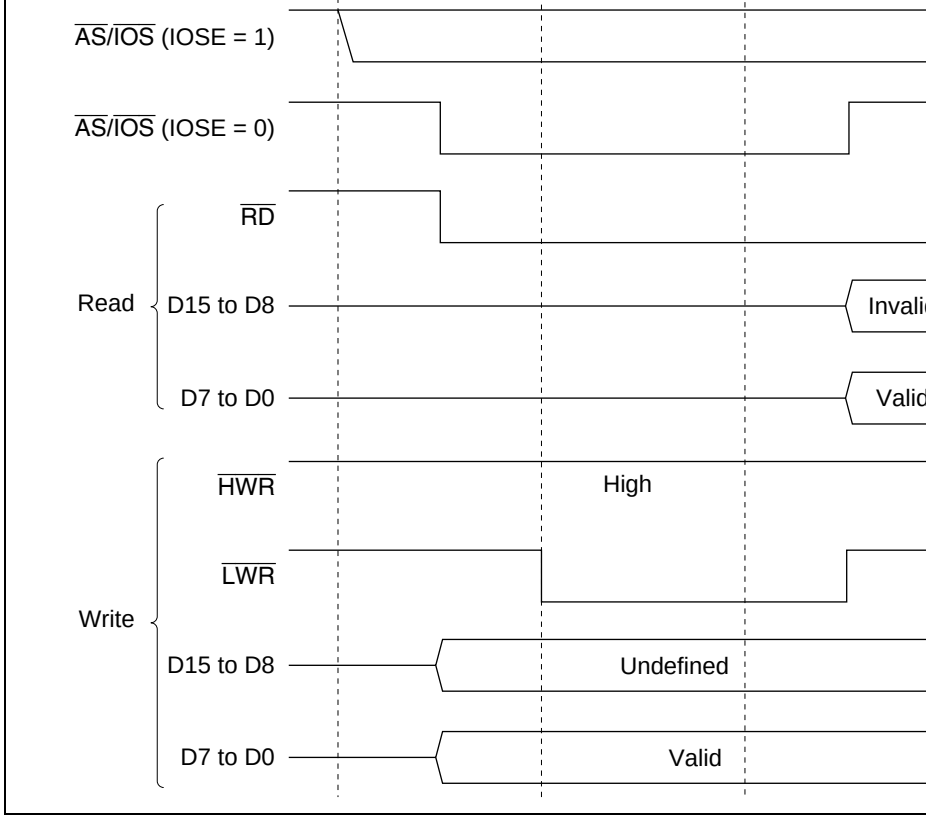


Figure 6.11 16-Bit, 3-State Access Space Bus Timing (2)
(Odd Address Byte Access)

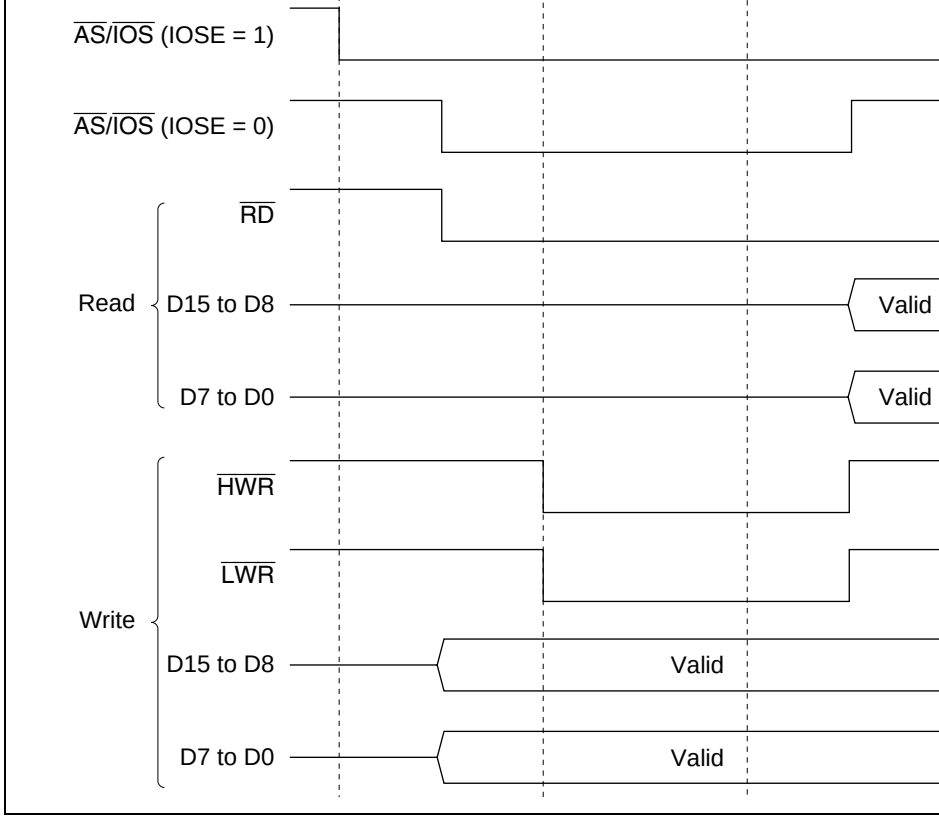


Figure 6.12 16-Bit, 3-State Access Space Bus Timing (3) (Word Access)

Pin Wait Mode: In pin wait mode, the number of T_w states specified by bits WC1 and
always inserted between the T_2 and T_3 states when external space is accessed. If the $\overline{W}$
low at the fall of ϕ in the last T_2 or T_w state, another T_w state is inserted. If the $\overline{WAIT}$ p
low, T_w states are inserted until it goes high.
Pin wait mode is useful for inserting four or more wait states, or for changing the num
states for different external devices.

Pin Auto-Wait Mode: In pin auto-wait mode, if the $\overline{WAIT}$ pin is low at the fall of ϕ i
state, the number of T_w states specified by bits WC1 and WC0 are inserted when exter
accessed. No additional T_w states are inserted even if the $\overline{WAIT}$ pin remains low. Pin
mode can be used for an easy interface to low-speed memory, simply by routing the c
signal to the $\overline{WAIT}$ pin.

Figure 6.13 shows an example of wait state insertion timing.

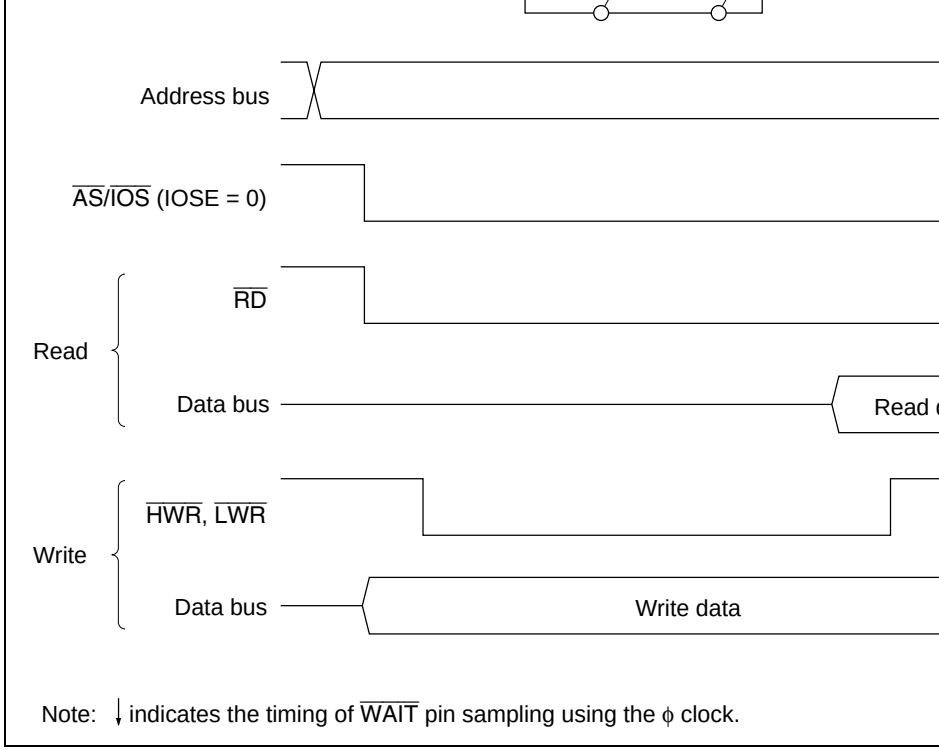


Figure 6.13 Example of Wait State Insertion Timing

The settings after a reset are: 3-state access, insertion of 3 program wait states, and WA
disabled.

Consecutive burst accesses of a maximum of 4 words or 8 words can be performed for instruction fetches only. One or two states can be selected for burst access.

6.5.2 Basic Timing

The number of states in the initial cycle (full access) of the burst ROM interface is in a with the setting of the AST bit. Also, when the AST bit is set to 1, wait state insertion One or two states can be selected for the burst cycle, according to the setting of the BR BCR. Wait states cannot be inserted.

When the BRSTS0 bit in BCR is cleared to 0, burst access of up to 4 words is perform the BRSTS0 bit is set to 1, burst access of up to 8 words is performed.

The basic access timing for burst ROM space is shown in figures 6.14 (a) and (b). The shown in figure 6.14 (a) is for the case where the AST and BRSTS1 bits are both set to in figure 6.14 (b) is for the case where both these bits are cleared to 0.

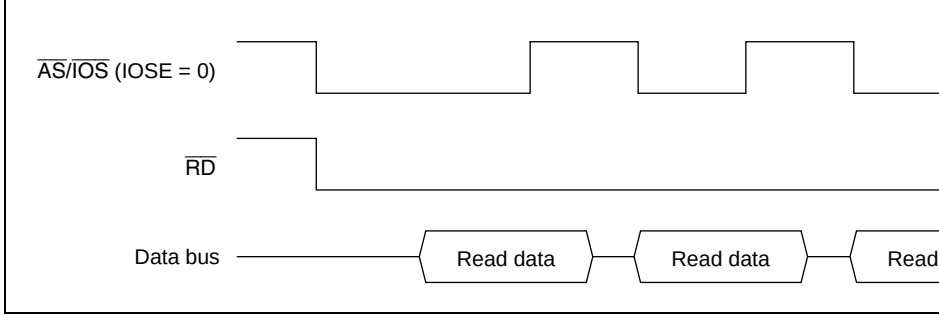


Figure 6.14 (a) Example of Burst ROM Access Timing (when AST = BRSTS

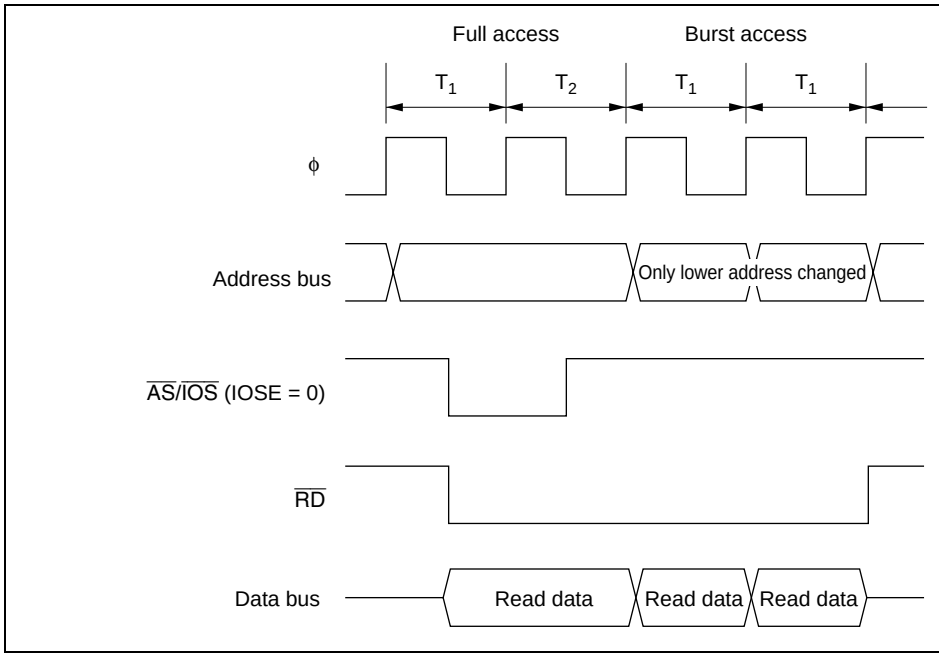


Figure 6.14 (b) Example of Burst ROM Access Timing (when AST = BRSTS

6.6 Idle Cycle

6.6.1 Operation

When this LSI chip accesses external space, it can insert a 1-state idle cycle (T_i) betwe
cycles when a write cycle occurs immediately after a read cycle. By inserting an idle c
possible, for example, to avoid data collisions between ROM, with a long output float
high-speed memory, I/O interfaces, and so on.

If an external write occurs after an external read while the ICIS0 bit in BCR is set to 1
cycle is inserted at the start of the write cycle. This is enabled in advanced mode and n
mode.

Figure 6.15 shows an example of the operation in this case. In this example, bus cycle
cycle from ROM with a long output floating time, and bus cycle B is a CPU write cyc
idle cycle is not inserted, and a collision occurs in cycle B between the read data from
the CPU write data. In (b), an idle cycle is inserted, and a data collision is prevented.

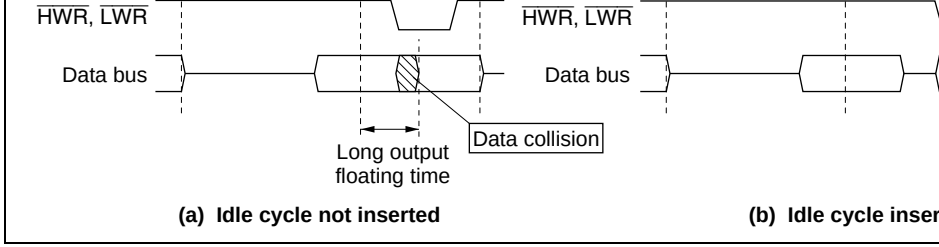


Figure 6.15 Example of Idle Cycle Operation

6.6.2 Pin States in Idle Cycle

Table 6.6 shows pin states in an idle cycle.

Table 6.6 Pin States in Idle Cycle

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

bus arbiter determines priorities at the prescribed timing, and permits use of the bus by
bus request acknowledge signal. The selected bus master then takes possession of the
begins its operation.

6.7.2 Operation

The bus arbiter detects the bus masters' bus request signals, and if the bus is requested
request acknowledge signal to the bus master making the request. If there are bus requ
both bus masters, the bus request acknowledge signal is sent to the one with the highe
When a bus master receives the bus request acknowledge signal, it takes possession of
until that signal is canceled.

The order of priority of the bus masters is as follows:

(High) DTC > CPU (Low)

The CPU is the lowest-priority bus master, and if a bus request is received from the DT arbiter transfers the bus to the DTC. The timing for transfer of the bus is as follows:

- The bus is transferred at a break between bus cycles. However, if a bus cycle is exe discrete operations, as in the case of a longword-size access, the bus is not transferr the operations.

  See appendix A.5, Bus States during Instruction Execution, for timings at which the transferred.
- If the CPU is in sleep mode, it transfers the bus immediately.

DTC

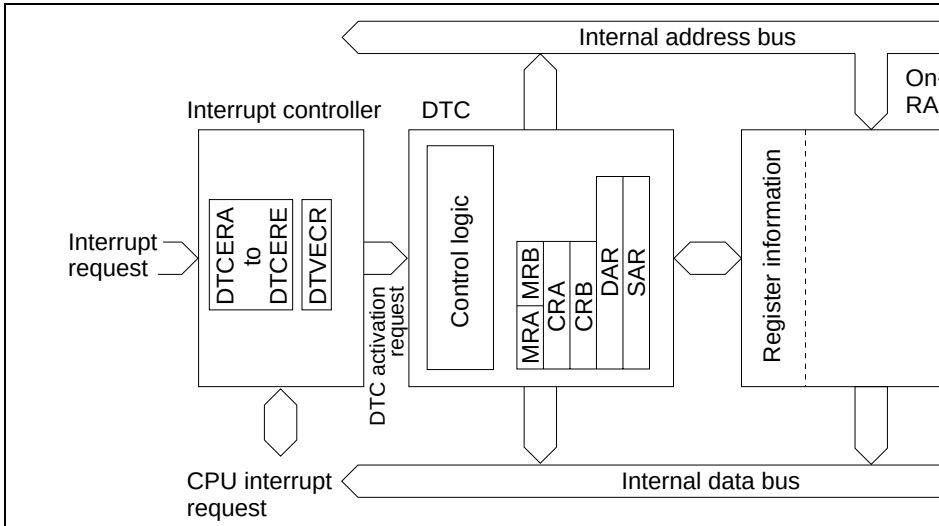
The DTC sends the bus arbiter a request for the bus when an activation request is gener

The DTC does not release the bus until it has completed a series of processing operatio

interrupt or software, to transfer data.

7.1.1 Features

- Transfer possible over any number of channels
  - Transfer information is stored in memory
  - One activation source can trigger a number of data transfers (chain transfer)
- Wide range of transfer modes
  - Normal, repeat, and block transfer modes available
  - Incrementing, decrementing, and fixing of transfer source and destination addr selected
- Direct specification of 16-Mbyte address space possible
  - 24-bit transfer source and destination addresses can be specified
- Transfer can be set in byte or word units
- A CPU interrupt can be requested for the interrupt that activated the DTC
  - An interrupt request can be issued to the CPU after one data transfer ends
  - An interrupt request can be issued to the CPU after all specified data transfers
- Activation by software is possible
- Module stop mode can be set
  - The initial setting enables DTC registers to be accessed. DTC operation is halt module stop mode



Legend:
MRA, MRB: DTC mode registers A and B
CRA, CRB: DTC transfer count registers A and B
SAR: DTC source address register
DAR: DTC destination address register
DTCERA to DTCERE: DTC enable registers A to E
DTVECR: DTC vector register

Figure 7.1 Block Diagram of DTC

DTC source address register	SAR	—*2	Undefined	—*3
DTC destination address register	DAR	—*2	Undefined	—*3
DTC transfer count register A	CRA	—*2	Undefined	—*3
DTC transfer count register B	CRB	—*2	Undefined	—*3
DTC enable registers	DTCER*4	R/W	H'00	H'FEEE
DTC vector register	DTVECR*4	R/W	H'00	H'FEF3
Module stop control register	MSTPCRH	R/W	H'3F	H'FF86
	MSTPCRL	R/W	H'FF	H'FF87

Notes: 1. Lower 16 bits of the address.

2. Registers within the DTC cannot be read or written to directly.

3. Allocated to on-chip RAM addresses H'EC00 to H'EFFF as register informa
They cannot be located in external memory space.

When the DTC is used, do not clear the RAME bit in SYSCR to 0.

4. The H8S/2144 Group and H8S/2147N do not include an on-chip DTC, and
DTCER and DTVECR register addresses should not be accessed by the C

Read/Write	—	—	—	—	—	—	—

MRA is an 8-bit register that controls the DTC operating mode.

Bits 7 and 6—Source Address Mode 1 and 0 (SM1, SM0): These bits specify whethe to be incremented, decremented, or left fixed after a data transfer.

Bit 7	Bit 6	
SM1	**SM0**	**Description**
0	—	SAR is fixed
1	0	SAR is incremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)
	1	SAR is decremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)

Bits 5 and 4—Destination Address Mode 1 and 0 (DM1, DM0): These bits specify v DAR is to be incremented, decremented, or left fixed after a data transfer.

Bit 5	Bit 4	
DM1	**DM0**	**Description**
0	—	DAR is fixed
1	0	DAR is incremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)
	1	DAR is decremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)

Bit 1—DTC Transfer Mode Select (DTS): Specifies whether the source side or the d
side is set to be a repeat area or block area, in repeat mode or block transfer mode.

Bit 1

DTS	Description
0	Destination side is repeat area or block area
1	Source side is repeat area or block area

Bit 0—DTC Data Transfer Size (Sz): Specifies the size of data to be transferred.

Bit 0

Sz	Description
0	Byte-size transfer
1	Word-size transfer

MRB is an 8-bit register that controls the DTC operating mode.

Bit 7—DTC Chain Transfer Enable (CHNE): Specifies chain transfer. In chain trans
multiple data transfers can be performed consecutively in response to a single transfer r
With data transfer for which CHNE is set to 1, there is no determination of the end of t
number of transfers, clearing of the interrupt source flag, or clearing of DTCER.

Bit 7

CHNE	Description
0	End of DTC data transfer (activation waiting state is entered)
1	DTC chain transfer (new register information is read, then data is transferre

Bit 6—DTC Interrupt Select (DISEL): Specifies whether interrupt requests to the CP
disabled or enabled after a data transfer.

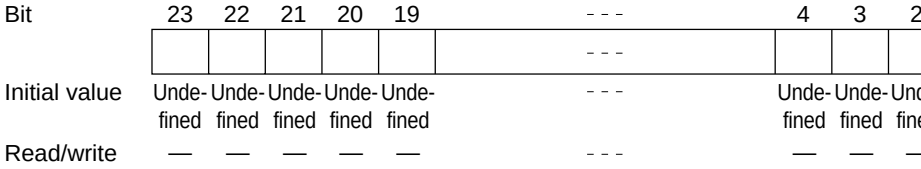
Bit 6

DISEL	Description
0	After a data transfer ends, the CPU interrupt is disabled unless the transfer 0 (the DTC clears the interrupt source flag of the activating interrupt to 0)
1	After a data transfer ends, the CPU interrupt is enabled (the DTC does not interrupt source flag of the activating interrupt to 0)

Bits 5 to 0—Reserved: In the H8S/2148 Group these bits have no effect on DTC opera
should always be written with 0.

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

CRA is a 16-bit register that designates the number of times data is to be transferred by

In normal mode, the entire CRA register functions as a 16-bit transfer counter (1 to 65,
decremented by 1 every time data is transferred, and transfer ends when the count reac

In repeat mode or block transfer mode, CRA is divided into two parts: the upper 8 bits
and the lower 8 bits (CRAL). CRAH holds the number of transfers while CRAL functi
8-bit transfer counter (1 to 256). CRAL is decremented by 1 every time data is transfer
contents of CRAH are transferred when the count reaches H'00. This operation is repea

7.2.6 DTC Transfer Count Register B (CRB)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined
Read/Write	—	—	—	—	—	—	—	—	—	—	—	—	—	—

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 65,536) that is decrer
every time data is transferred, and transfer ends when the count reaches H'0000.

with bits corresponding to the interrupt sources that can activate the DTC. These bits e
disable DTC service for the corresponding interrupt sources.

The DTC enable registers are initialized to H'00 by a reset and in hardware standby m

Bit n—DTC Activation Enable (DTCEn)

Bit n

DTCEn	Description
0	DTC activation by interrupt is disabled ([Clearing conditions] • When data transfer ends with the DISEL bit set to 1 • When the specified number of transfers end
1	DTC activation by interrupt is enabled [Holding condition] When the DISEL bit is 0 and the specified number of transfers have not er

Note: n = 7 to 0

A DTCE bit can be set for each interrupt source that can activate the DTC. The corres
between interrupt sources and DTCE bits is shown in table 7.4, together with the vecto
generated by the interrupt controller in each case.

For DTCE bit setting, read/write operations must be performed using bit-manipulation
such as BSET and BCLR. For the initial setting only, however, when multiple activati
are set at one time, it is possible to disable interrupts and write after executing a dumn
the relevant register.

is read.

DTVECR is an 8-bit readable/writable register that enables or disables DTC activation software, and sets a vector number for the software activation interrupt.

DTVECR is initialized to H'00 by a reset and in hardware standby mode.

Bit 7—DTC Software Activation Enable (SWDTE): Specifies enabling or disabling software activation. To clear the SWDTE bit by software, read SWDTE when set to 1, in the bit.

Bit 7

SWDTE	Description
0	DTC software activation is disabled (Ini [Clearing condition] When the DISEL bit is 0 and the specified number of transfers have not e
1	DTC software activation is enabled [Holding conditions] • When data transfer ends with the DISEL bit set to 1 • When the specified number of transfers end • During software-activated data transfer

Bits 6 to 0—DTC Software Activation Vectors 6 to 0 (DTVEC6 to DTVEC0): Thes specify a vector number for DTC software activation.

The vector address is H'0400 + (vector number) << 1 (where << 1 indicates a 1-bit left example, if DTVEC6 to DTVEC0 = H'10, the vector address is H'0420.

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mo

When the MSTP14 bit in MSTPCR is set to 1, the DTC operation stops at the end of t and a transition is made to module stop mode. Note that 1 cannot be written to the MS when the DTC is being activated. For details, see section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not in software standby mode.

MSTPCRH Bit 6—Module Stop (MSTP14): Specifies the DTC module stop mode.

MSTPCRH Bit 6	
MSTP14	**Description**
0	DTC module stop mode is cleared
1	DTC module stop mode is set

7.3 Operation

7.3.1 Overview

When activated, the DTC reads register information that is already stored in memory a data on the basis of that register information. After the data transfer, it writes updated information back to memory. Pre-storage of register information in memory makes it transfer data over any required number of channels. Setting the CHNE bit to 1 makes perform a number of transfers with a single activation.

Figure 7.2 shows a flowchart of DTC operation.

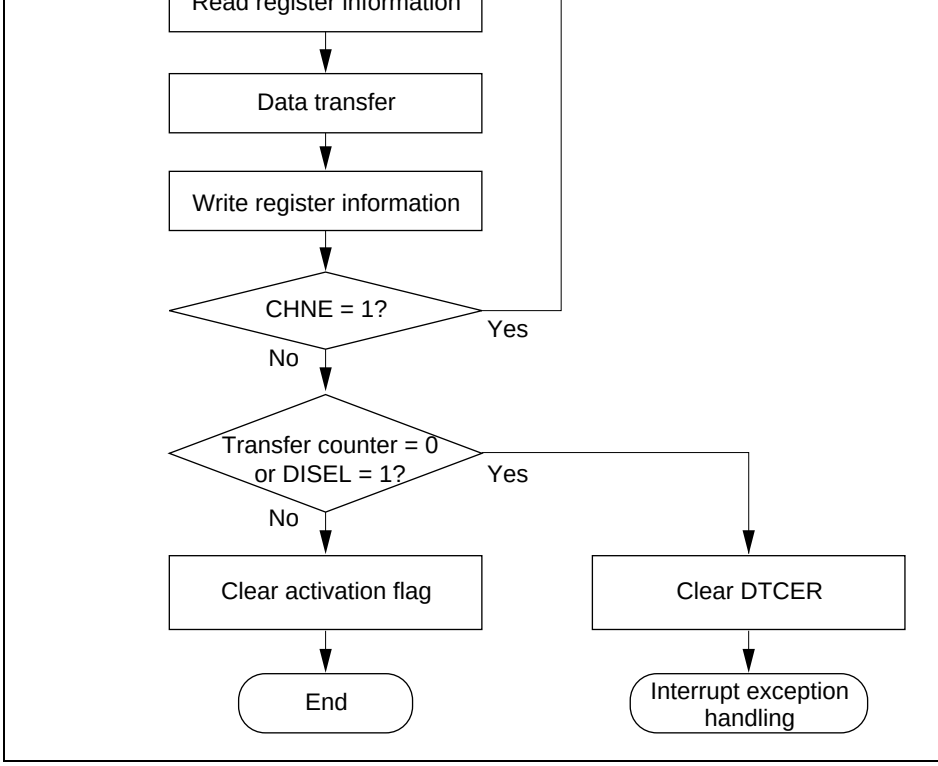


Figure 7.2 Flowchart of DTC Operation

The DTC transfer mode can be normal mode, repeat mode, or block transfer mode.

The 24-bit SAR designates the DTC transfer source address and the 24-bit DAR design transfer destination address. After each transfer, SAR and DAR are independently incre decremented, or left fixed.

Table 7.2 outlines the functions of the DTC.

or decremented by 1 or 2
— Up to 65,536 transfers possible
- Repeat mode
— One transfer request transfers one byte or one word
— Memory addresses are incremented or decremented by 1 or 2
— After the specified number of transfers (1 to 256), the initial state resumes and operation continues
- Block transfer mode
— One transfer request transfers a block of the specified size
— Block size is from 1 to 256 bytes or words
— Up to 65,536 transfers possible
— A block area can be designated at either the source or destination

- Host interface IBF
- SCI TXI or RXI
- A/D converter ADI
- IIC IICI
- Software

7.3.2 Activation Sources

The DTC operates when activated by an interrupt or by a write to DTVECR by softwa
activation). An interrupt request can be directed to the CPU or DTC, as designated by
corresponding DTCER bit. The interrupt request is directed to the DTC when the corr
bit is set to 1, and to the CPU when the bit is cleared to 0.

At the end of one data transfer (or the last of the consecutive transfers in the case of cl
the interrupt source or the corresponding DTCER bit is cleared. Table 7.3 shows activ
sources and DTCER clearing.

Interrupt activation	• Corresponding DTCER bit held at 1 • Activation source flag cleared to 0	• Corresponding DTCER bit c • Activation source flag held a • Activation source interrupt r sent to CPU

Figure 7.3 shows a block diagram of activation source control. For details see section 5 Controller.

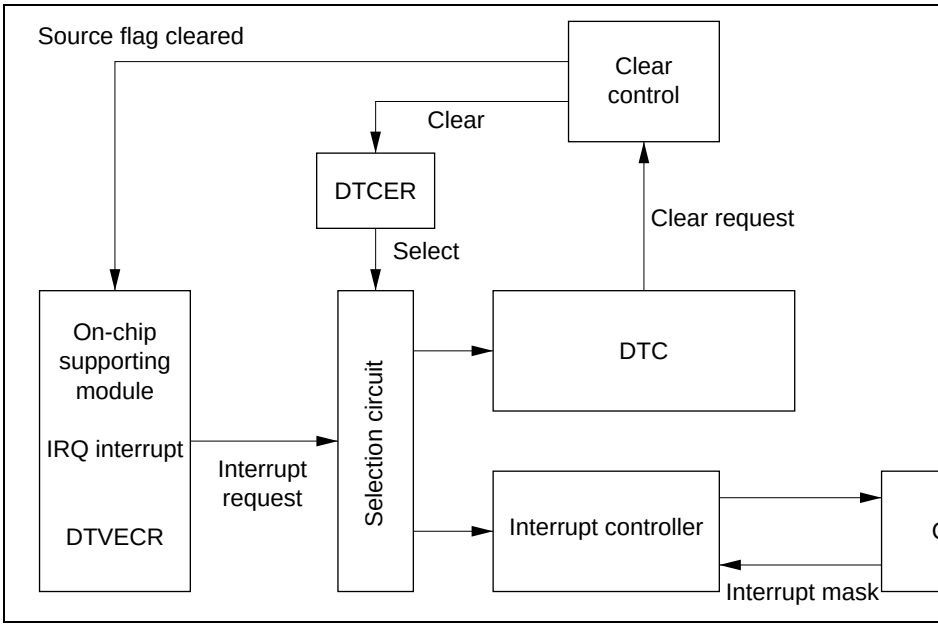


Figure 7.3 Block Diagram of DTC Activation Source Control

When an interrupt has been designated a DTC activation source, existing CPU mask lev interrupt controller priorities have no effect. If there is more than one activation source time, the DTC is activated in accordance with the default priorities.

The DTC reads the start address of the register information from the vector address se
activation source, and then reads the register information from that start address. The r
information can be placed at predetermined addresses in the on-chip RAM. The start a
the register information should be an integral multiple of four.

The configuration of the vector address is the same in both normal and advanced mod
unit being used in both cases. These two bytes specify the lower bits of the address in
RAM.

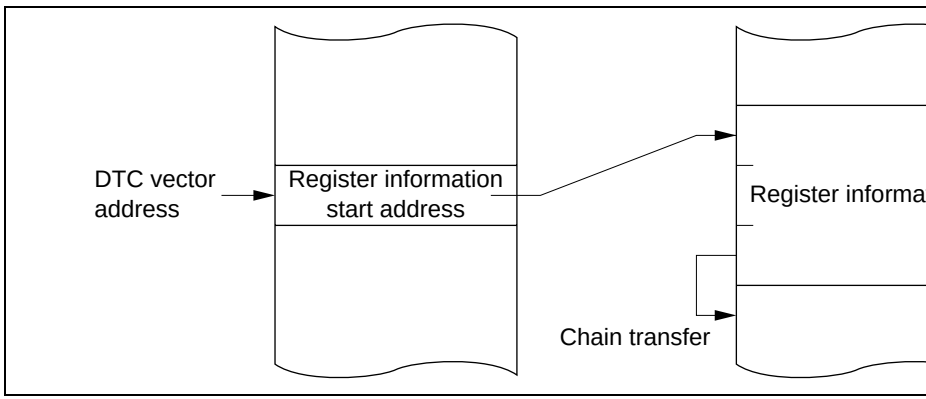


Figure 7.4 Correspondence between DTC Vector Address and Register Info

IRQ2		18	H'0424	DTCEA
IRQ3		19	H'0426	DTCEA
ADI (A/D conversion end)	A/D	28	H'0438	DTCEA
ICIA (FRT input capture A)	FRT	48	H'0460	DTCEA
ICIB (FRT input capture B)		49	H'0462	DTCEA
OCIA (FRT output compare A)		52	H'0468	DTCEA
OCIB (FRT output compare B)		54	H'046A	DTCEB
CMIA0 (TMR0 compare-match A)	TMR0	64	H'0480	DTCEB
CMIB0 (TMR0 compare-match B)		65	H'0482	DTCEB
CMIA1 (TMR1 compare-match A)	TMR1	68	H'0488	DTCEB
CMIB1 (TMR1 compare-match B)		69	H'048A	DTCEC
CMIAY (TMRY compare-match A)	TMRY	72	H'0490	DTCEC
CMIBY (TMRY compare-match B)		73	H'0492	DTCEC
IBF1 (IDR1 reception completed)	HIF	76	H'0498	DTCEC
IBF2 (IDR2 reception completed)		77	H'049A	DTCEC
RXI0 (reception completed 0)	SCI channel 0	81	H'04A2	DTCEC
TXI0 (transmit data empty 0)		82	H'04A4	DTCEC
RXI1 (reception completed 1)	SCI channel 1	85	H'04AA	DTCEC
TXI1 (transmit data empty 1)		86	H'04AC	DTCED
RXI2 (reception completed 2)	SCI channel 2	89	H'04B2	DTCED
TXI2 (transmit data empty 2)		90	H'04B4	DTCED
IICI0 (IIC0 1-byte transmission/ reception completed)	IIC0 (option)	92	H'04B8	DTCED
IICI1 (IIC1 1-byte transmission/ reception completed)	IIC1 (option)	94	H'04BC	DTCED

Note: * DTCE bits with no corresponding interrupt are reserved, and should be writte

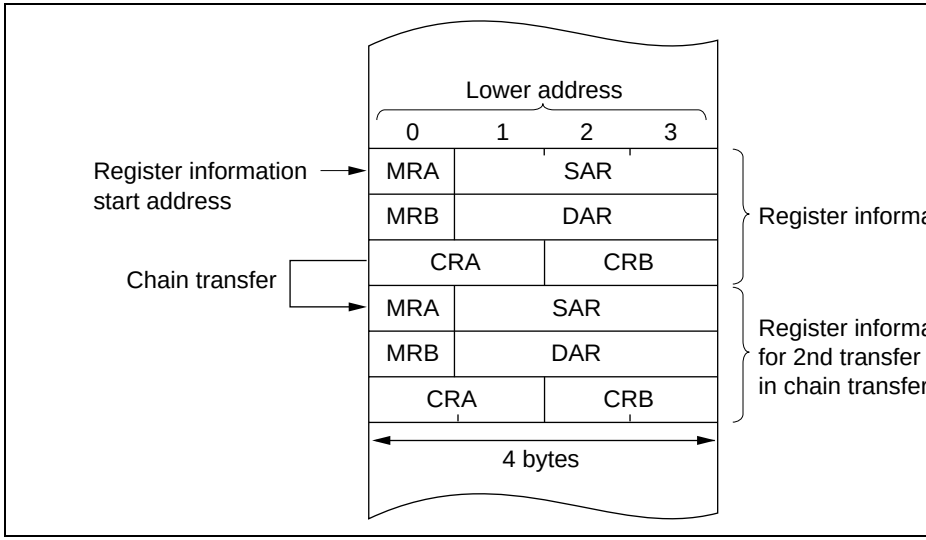


Figure 7.5 Location of DTC Register Information in Address Space

Table 7.5 Register Information in Normal Mode

Name	Abbreviation	Function
DTC source address register	SAR	Transfer source address
DTC destination address register	DAR	Transfer destination addre
DTC transfer count register A	CRA	Transfer count
DTC transfer count register B	CRB	Not used

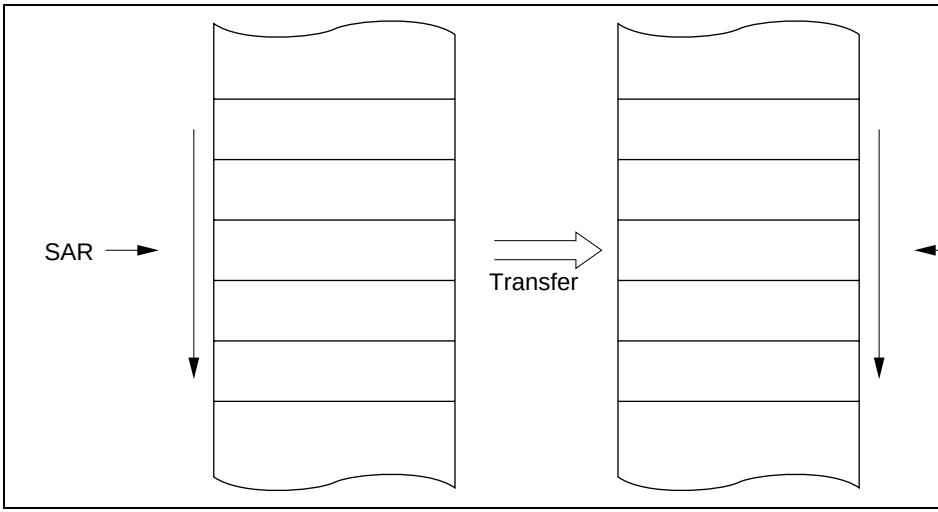


Figure 7.6 Memory Mapping in Normal Mode

Table 7.6 lists the register information in repeat mode and figure 7.7 shows memory m
repeat mode.

Table 7.6 Register Information in Repeat Mode

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

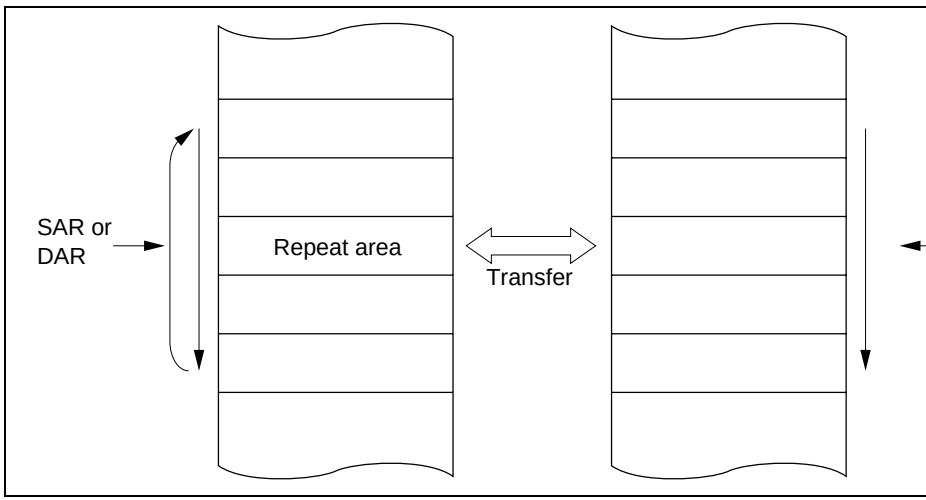


Figure 7.7 Memory Mapping in Repeat Mode

From 1 to 65,536 transfers can be specified. Once the specified number of transfers hav
CPU interrupt is requested.

Table 7.7 lists the register information in block transfer mode and figure 7.8 shows mer
mapping in block transfer mode.

Table 7.7 Register Information in Block Transfer Mode

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

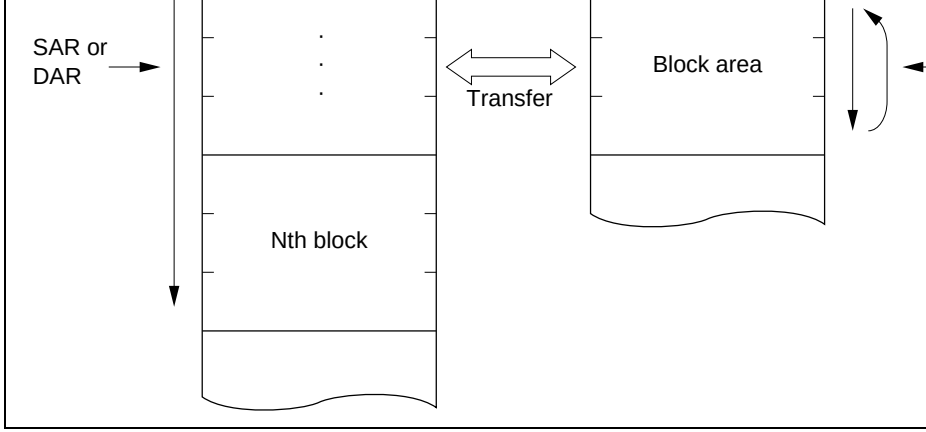


Figure 7.8 Memory Mapping in Block Transfer Mode

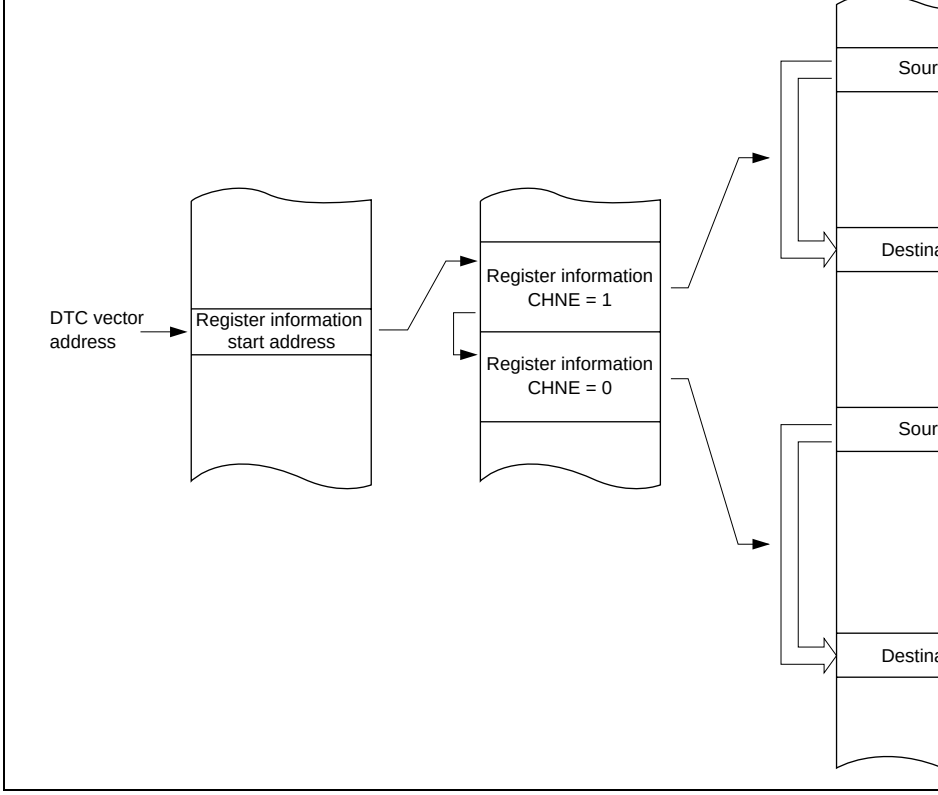


Figure 7.9 Memory Mapping in Chain Transfer

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

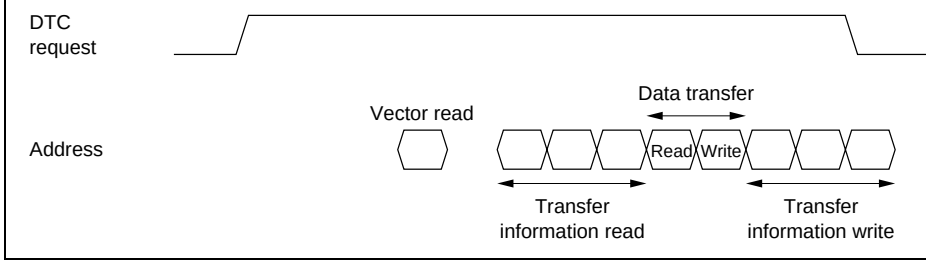


Figure 7.10 DTC Operation Timing (Normal Mode or Repeat Mode)

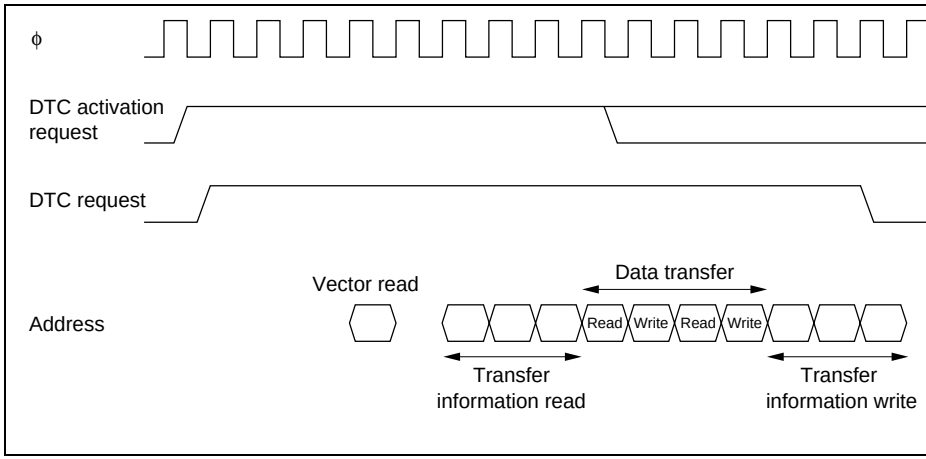


Figure 7.11 DTC Operation Timing (Block Transfer Mode, with Block Siz

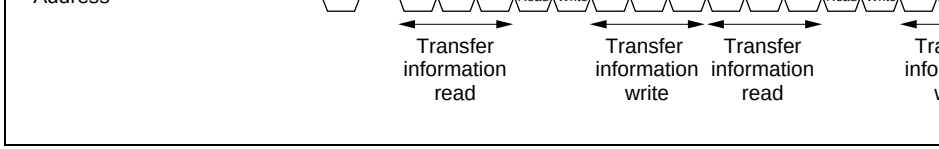


Figure 7.12 DTC Operation Timing (Chain Transfer)

7.3.10 Number of DTC Execution States

Table 7.8 lists execution phases for a single DTC data transfer, and table 7.9 shows the states required for each execution phase.

Table 7.8 DTC Execution Phases

Mode	Vector Read I	Register Information Read/Write J	Data Read K	Data Write L	In Op M
Normal	1	6	1	1	3
Repeat	1	6	1	1	3
Block transfer	1	6	N	N	3

Legend:

N: Block size (initial setting of CRAH and CRAL)

	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	1	1	1	1	1	1

The number of execution states is calculated from the formula below. Note that Σ mea
of all transfers activated by one activation event (the number for which the CHNE bit
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 r
and data is transferred from the on-chip ROM to an internal I/O register, the time requ
DTC operation is 13 states. The time from activation to the end of the data write is 10

3. Set the corresponding bit in DTCER to 1.

4. Set the enable bits for the interrupt sources to be used as the activation sources to 1. is activated when an interrupt used as an activation source is generated.

5. After the end of one data transfer, or after the specified number of data transfers hav the DTCE bit is cleared to 0 and a CPU interrupt is requested. If the DTC is to cont transferring data, set the DTCE bit to 1.

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 the on-chi

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 in the SWDTE bit and the vector number to DTVECR.

5. Check the vector number written to DTVECR.

6. After the end of one data transfer, if the DISEL bit is 0 and a CPU interrupt is not re the SWDTE bit is cleared to 0. If the DTC is to continue transferring data, set the S to 1. When the DISEL bit is 1, or after the specified number of data transfers have e SWDTE bit is held at 1 and a CPU interrupt is requested.

SCI RDR address in SAR, the start address of the RAM area where the data will b
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
complete (RXI) interrupt. Since the generation of a receive error during the SCI re
operation will disable subsequent reception, the CPU should be enabled to accept
interrupts.

5. Each time reception of one byte of data ends on the SCI, the RDRF flag in SSR is
RXI interrupt is generated, and the DTC is activated. The receive data is transferre
to RAM by the DTC. DAR is incremented and CRA is decremented. The RDRF fl
automatically cleared to 0.

6. When CRA becomes 0 after the 128 data transfers have ended, the RDRF flag is h
DTCE bit is cleared to 0, and an RXI interrupt request is sent to the CPU. The inte
handling routine should perform wrap-up processing.

0). Set the transfer source address (H'1000) in SAR, the destination address (H'2000
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 transf
by software.

4. Write 1 to the SWDTE bit and the vector number (H'60) to DTVECR. The write da

5. Read DTVECR again and check that it is set to the vector number (H'60). If it is no
indicates that the write failed. This is presumably because an interrupt occurred betv
3 and 4 and led to a different software activation. To activate this transfer, go back t

6. If the write was successful, the DTC is activated and a block of 128 bytes of data is
7. After the transfer, an SWDTEND interrupt occurs. The interrupt handling routine sh
the SWDTE bit to 0 and perform other wrap-up processing.

When the DISEL bit is 1 and one data transfer has ended, or the specified number of t have ended, after data transfer ends, the SWDTE bit is held at 1 and an SWDTEND in generated. The interrupt handling routine should clear the SWDTE bit to 0.

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

7.5 Usage Notes

Module Stop

When the MSTP14 bit in MSTPCR is set to 1, the DTC clock stops, and the DTC ente module stop state. However, 1 cannot be written in the MSTP14 bit while the DTC is When the DTC is placed in the module stop state, the DTCER registers must all be in state when the MSTP14 bit is set to 1.

On-Chip RAM

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

DTCE Bit Setting

For DTCE bit setting, read/write operations must be performed using bit-manipulation such as BSET and BCLR. For the initial setting only, however, when multiple activati are set at one time, it is possible to disable interrupts and write after executing a dumm the relevant register.

Each port includes a data direction register (DDR) that controls input/output (not prov
input-only port) and data registers (DR, ODR) that store output data.

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

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, 2, and 3 can driv
(10 mA sink current).

PA4 to PA7 in port A have bus buffer drive capability. In the H8S/2148 Group and H
P52 in port 5 and P97 in port 9 are NMOS push-pull outputs.

Note that the H8S/2144 Group and H8S/2147N have subset specifications that do not
some supporting modules. For differences in pin functions, see table 8.1, H8S/2148 G
Functions, table 8.2, H8S/2147N Port Functions, and table 8.3, H8S/2144 Group Port

	• LED drive capability			lower address output (A7 to A0) or PWM timer output (PW7 to PW0)	
Port 2	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P27/A15/PW15/ CBLANK P26/A14/PW14 P25/A13/PW13 P24/A12/PW12 P23/A11/PW11 P22/A10/PW10 P21/A9/PW9 P20/A8/PW8	Upper address output (A15 to A8)	When DDR = 0 (after reset): input port or timer connection output (CBLANK) When DDR = 1: upper address output (A15 to A8), PWM timer output (PW15 to PW8), timer connection output (CBLANK), or output ports (P27 to P24)	I/O port also fu PWM timer ou to PW8) and t connection ou (CBLANK)
Port 3	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P37 to P30/ HDB7 to HDB0/ D15 to D8	Data bus input/output (D15 to D8)		I/O port also fu host interface input/output (H HDB0)

		P43/TMCI1/ HIRQ11/HSYNCI P42/TMRI0/ SCK2/SDA1 P41/TMO0/ RxD2/IrRxD P40/TMCI0/ TxD2/IrTxD	RxD2, SCK2), IrDA interface input/output (IrTxD, IrRxD), and I²C bus interface 1 (option) input/output (SDA1)	input/output CSYNCI, HS interface hos interrupt requ (HIRQ12, HI HIRQ11), SC output (TxD2 SCK2), IrDA input/output and I²C bus i (option) inpu (SDA1)
Port 5	• 3-bit I/O port	P52/SCK0/SCL0 P51/RxD0 P50/TxD0	I/O port also functioning as SCI0 input/output (TxD SCK0) and I²C bus interface 0 (option) input/output	
Port 6	• 8-bit I/O port	P67/$\overline{\text{IRQ7}}$/TMOX/ $\overline{\text{KIN7}}$/CIN7 P66/$\overline{\text{IRQ6}}$/FTOB/ $\overline{\text{KIN6}}$/CIN6 P65/FTID/$\overline{\text{KIN5}}$/ CIN5 P64/FTIC/$\overline{\text{KIN4}}$/ CIN4/CLAMPO P63/FTIB/$\overline{\text{KIN3}}$/ CIN3/VFBACKI P62/FTIA/TMIY/ $\overline{\text{KIN2}}$/CIN2/ VSYNCI P61/FTOA/$\overline{\text{KIN1}}$/ CIN1/VSYNCO P60/FTCI/TMIX/ $\overline{\text{KIN0}}$/CIN0/ HFBACKI	I/O port also functioning as external interrupt input FRT input/output (FTCI, FTOA, FTIA, FTIB, FTIC, 8-bit timer X and Y input/output (TMOX, TMIX, TM connection input/output (CLAMPO, VFBACKI, VSY VSYNCO, HFBACKI), key-sense interrupt input ($\overline{\text{K}}$ and expansion A/D converter input (CIN7 to CIN0)	

		P72/AN2 P71/AN1 P70/AN0		
Port 8	• 7-bit I/O port	P86/IRQ5/SCK1/SCL1 P85/IRQ4/RxD1 P84/IRQ3/TxD1 P83 P82/HIFSD P81/CS2/GA20 P80/HA0	I/O port also functioning as external interrupt input (IRQ5, IRQ4, IRQ3), SCI1 input/output (TxD1, RxD1, SCK1), and I²C bus interface 1 (option) input/output (SCL1)	I/O port also fu external interr (IRQ5, IRQ4, input/output (T SCK1), host in control input/o GA20, HA0, H I²C bus interfa input/output (S
Port 9	• 8-bit I/O port	P97/WAIT/SDA0	I/O port also functioning as expanded data bus control input (WAIT) and I²C bus interface 0 (option) input/output (SDA0)	I/O port also fu I²C bus interfa input/output (S
		P96/φ/EXCL	When DDR = 0: input port or EXCL input When DDR = 1 (after reset): φ output	When DDR = 0 (after reset): input p input When DDR = 1: φ output
		P95/AS/IOS/CS1 P94/HWR/IOW P93/RD/IOR	Expanded data bus control output (AS/IOS, HWR, RD)	I/O port also fun host interface co (CS1, IOW, IOR

			trigger input (ADTRG)		(ADTRG), an interface con ($\overline{ECS2}$)
Port A	• 8-bit I/O port	PA7/A23/$\overline{KIN15}$/ CIN15/PS2CD PA6/A22/$\overline{KIN14}$/ CIN14/PS2CC PA5/A21/$\overline{KIN13}$/ CIN13/PS2BD PA4/A20/$\overline{KIN12}$/ CIN12/PS2BC PA3/A19/$\overline{KIN11}$/ CIN11/PS2AD PA2/A18/$\overline{KIN10}$/ CIN10/PS2AC PA1/A17/$\overline{KIN9}$/ CIN9 PA0/A16/$\overline{KIN8}$/ CIN8	I/O port also functioning as key-sense interrupt input ($\overline{KIN15}$ to $\overline{KIN8}$), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also functioning as address output (A23 to A16), key-sense interrupt input ($\overline{KIN15}$ to $\overline{KIN8}$), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/ output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also key-sense in ($\overline{KIN15}$ to $\overline{KI}$ expansion A input (CIN15 keyboard bu input/output PS2CC, PS2 PS2AD, PS2
Port B	• 8-bit I/O port	PB7/D7 PB6/D6 PB5/D5 PB4/D4 PB3/D3/$\overline{CS4}$ PB2/D2/$\overline{CS3}$ PB1/D1/HIRQ4 PB0/D0/HIRQ3	In 8-bit bus mode (ABW = 1): I/O port In 16-bit bus mode (ABW = 0): data bus input/output (D7 to D0)		I/O port also HIF control i pins ($\overline{CS3}$, $\overline{C}$ HIRQ4)

	• LED drive capability			output (A7 to A0) or PWM timer output (PW7 to PW0)	
Port 2	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P27/A15/PW15 P26/A14/PW14 P25/A13/PW13 P24/A12/PW12 P23/A11/PW11 P22/A10/PW10 P21/A9/PW9 P20/A8/PW8	Upper address output (A15 to A8)	When DDR = 0 (after reset): input port When DDR = 1: upper address output (A15 to A8), PWM timer output (PW15 to PW8), or output ports (P27 to P24)	I/O port also fu PWM timer ou to PW8)
Port 3	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P37 to P30/ HDB7 to HDB0/ D15 to D8	Data bus input/output (D15 to D8)		I/O port also fu host interface input/output (H HDB0)
Port 4	• 8-bit I/O port	P47/PWX1 P46/PWX0 P45/TMRI1/ HIRQ12 P44/TMO1/ HIRQ1 P43/TMCI1/ HIRQ11 P42/TMRI0/ SCK2/SDA1 P41/TMO0/ RxD2/IrRxD P40/TMCI0/ TxD2/IrTxD	I/O port also functioning as 14-bit PWM timer output (PWX1, PWX0), 8-bit timer 0 and 1 input/output (TMCI0, TMRI0, TMO0, TMCI1, TMRI1, TMO1), SCI2 input/output (TxD2, RxD2, SCK2), IrDA interface input/output (IrTxD, IrRxD), and I^2C bus interface 1 (option) input/output (SDA1)		I/O port also fu 14-bit PWM tir (PWX1, PWX 0 and 1 input/ (TMCI0, TMRI TMCI1, TMRI host interface interrupt reque (HIRQ12, HIR HIRQ11), SCI output (TxD2, SCK2), IrDA i input/output (I and I^2C bus in (option) input/ (SDA1)

		P66/IRQ6/FTOB/KIN6/CIN6 P65/FTID/KIN5/CIN5 P64/FTIC/KIN4/CIN4 P63/FTIB/KIN3/CIN3 P62/FTIA/TMIY/KIN2/CIN2 P61/FTOA/KIN1/CIN1 P60/FTCI/KIN0/CIN0	KIN0), and expansion A/D converter input (CIN7 to	
Port 7	• 8-bit I/O port	P77/AN7/DA1 P76/AN6/DA0 P75/AN5 P74/AN4 P73/AN3 P72/AN2 P71/AN1 P70/AN0	Input port also functioning as A/D converter analog AN0) and D/A converter analog output (DA1, DA0)	
Port 8	• 7-bit I/O port	P86/IRQ5/SCK1/SCL1 P85/IRQ4/RxD1 P84/IRQ3/TxD1 P83 P82/HIFSD P81/CS2/GA20 P80/HA0	I/O port also functioning as external interrupt input (IRQ5, IRQ4, IRQ3), SCI1 input/output (TxD1, RxD1, SCK1), and I²C bus interface 1 (option) input/output (SCL1)	I/O port also external inte (IRQ5, IRQ4 input/output SCK1), host control input/ GA20, HA0, I²C bus interf input/output

			EXCL input When DDR = 1 (after reset): φ output	When DDR = 1: φ output	
		P95/AS/IOS/CS1 P94/HWR/IOW P93/RD/IOR	Expanded data bus control output (AS/IOS, HWR, RD)		I/O port also fu host interface (CS1, IOW, IC
		P92/IRQ0 P91/IRQ1	I/O port also functioning as external interrupt input (I		
		P90/LWR/IRQ2/ ADTRG/ECS2	I/O port also functioning as expanded data bus control output (LWR), external interrupt input (IRQ2), and A/D converter external trigger input (ADTRG)		I/O port also fu external interr (IRQ2), A/D co external trigge (ADTRG), and interface contr (ECS2)
Port A	• 8-bit I/O port	PA7/A23/KIN15/ CIN15/PS2CD PA6/A22/KIN14/ CIN14/PS2CC PA5/A21/KIN13/ CIN13/PS2BD PA4/A20/KIN12/ CIN12/PS2BC PA3/A19/KIN11/ CIN11/PS2AD PA2/A18/KIN10/ CIN10/PS2AC PA1/A17/KIN9/ CIN9 PA0/A16/KIN8/ CIN8	I/O port also functioning as key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also functioning as address output (A23 to A16), key-sense interrupt input (KIN15 to KIN8), expansion A/D converter input (CIN15 to CIN8), and keyboard buffer controller input/ output (PS2CD, PS2CC, PS2BD, PS2BC, PS2AD, PS2AC)	I/O port also fu key-sense inte (KIN15 to KIN A/D converter to CIN8), and buffer controll input/output (P PS2CC, PS2B PS2AD, PS2A

		PB2/D2/CS3 PB1/D1/HIRQ4 PB0/D0/HIRQ3		

	• LED drive capability			output (A7 to A0)	
Port 2	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P27 to P20/ A15 to A8	Upper address output (A15 to A8)	When DDR = 0 (after reset): input port When DDR = 1: upper address output (A15 to A8) or output port (P27 to P24)	I/O port
Port 3	• 8-bit I/O port • Built-in MOS input pull-ups • LED drive capability	P37 to P30/ D15 to D8	Data bus input/output (D15 to D8)		I/O port
Port 4	• 8-bit I/O port	P47/PWX1 P46/PWX0 P45/TMRI1 P44/TMO1 P43/TMCI1 P42/TMRI0/SCK2 P41/TMO0/RxD2/ IrRxD P40/TMCI0/ TxD2/IrTxD	I/O port also functioning as 14-bit PWM timer output PWX0), 8-bit timer 0 and 1 input/output (TMCI0, TM TMCI1, TMRI1, TMO1), SCI2 input/output (TxD2, Rx and IrDA interface input/output (IrTxD, IrRxD)		
Port 5	• 3-bit I/O port	P52/SCK0 P51/RxD0 P50/TxD0	I/O port also functioning as SCI0 input/output (TxD0 SCK0)		

		P64/FTIC/$\overline{\text{KIN4}}$/CIN4 P63/FTIB/$\overline{\text{KIN3}}$/CIN3 P62/FTIA/TMIY/$\overline{\text{KIN2}}$/CIN2 P61/FTOA/$\overline{\text{KIN1}}$/CIN1 P60/FTCI/$\overline{\text{KIN0}}$/CIN0	
Port 7	• 8-bit input port	P77/AN7/DA1 P76/AN6/DA0 P75/AN5 P74/AN4 P73/AN3 P72/AN2 P71/AN1 P70/AN0	Input port also functioning as A/D converter analog AN0) and D/A converter analog output (DA1, DA0)
Port 8	• 7-bit I/O port	P86/$\overline{\text{IRQ5}}$/SCK1 P85/$\overline{\text{IRQ4}}$/RxD1 P84/$\overline{\text{IRQ3}}$/TxD1 P83 P82 P81 P80	I/O port also functioning as external interrupt input $\overline{\text{IRQ3}}$) and SCI1 input/output (TxD1, RxD1, SCK1)

			When DDR = 1 (after reset): φ output		
		P95/AS/IOS P94/HWR P93/RD	Expanded data bus control output(AS/IOS, HWR, RD)		I/O port
		P92/IRQ0 P91/IRQ1	I/O port also functioning as external interrupt input (I		
		P90/LWR/ IRQ2/ADTRG	I/O port also functioning as expanded data bus control output (LWR), external interrupt input (IRQ2), and A/D converter external trigger input (ADTRG)		I/O port also fu external interr (IRQ2) and A/ external trigge (ADTRG)
Port A	• 8-bit I/O port	PA7 to PA0/ A23 to A16/ KIN15 to KIN8/ CIN15 to CIN8	I/O port also functioning as key-sense interrupt input (KIN15 to KIN8), and expansion A/D converter input (CIN15 to CIN8)	I/O port also functioning as address output (A23 to A16), key-sense interrupt input (KIN15 to KIN8), and expansion A/D converter input (CIN15 to CIN8)	I/O port also fu key-sense inte (KIN15 to KIN expansion A/D input (CIN15 t
Port B	• 8-bit I/O port	PB7 to PB0/ D7 to D0	In 8-bit bus mode (ABW = 1): I/O port In 16-bit bus mode (ABW = 0): data bus input/output (D7 to D0)		I/O port

Figure 8.1 shows the port 1 pin configuration.

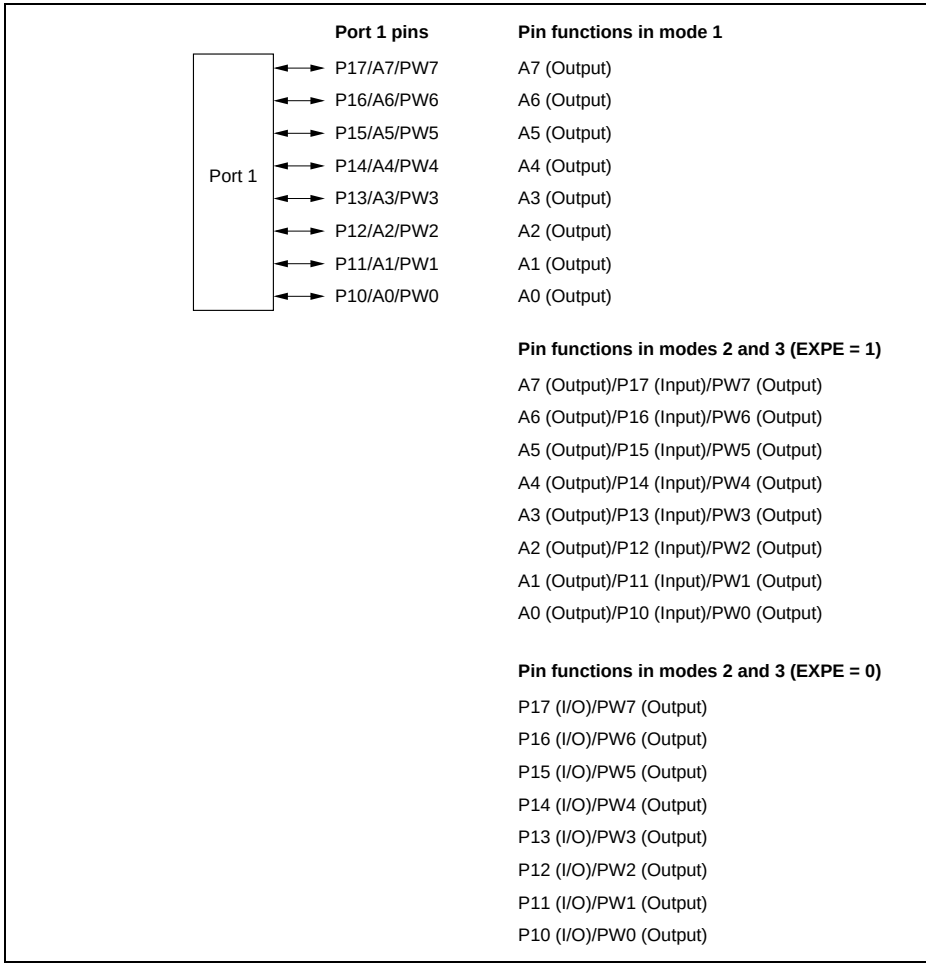


Figure 8.1 Port 1 Pin Functions

Port 1 MOS pull-up control register	P1PCR	R/W	H'00	H'FF

Note: * Lower 16 bits of the address.

Port 1 Data Direction Register (P1DDR)

Bit	7	6	5	4	3	2	1	
	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

P1DDR is an 8-bit write-only register, the individual bits of which specify input or outp pins of port 1. P1DDR cannot be read; if it is, an undefined value will be returned.

P1DDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri software standby mode. The address output pins maintain their output state in a transiti software standby mode.

- Mode 1

  The corresponding port 1 pins are address outputs, regardless of the P1DDR setting

  In hardware standby mode, the address outputs go to the high-impedance state.

- Modes 2 and 3 (EXPE = 1)

  The corresponding port 1 pins are address outputs or PWM outputs when P1DDR b to 1, and input ports when cleared to 0.

- Modes 2 and 3 (EXPE = 0)

  The corresponding port 1 pins are output ports or PWM outputs when P1DDR bits a and input ports when cleared to 0.

If a port 1 read is performed while P1DDR bits are set to 1, the P1DR values are read regardless of the actual pin states. If a port 1 read is performed while P1DDR bits are the pin states are read.

P1DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prio software standby mode.

Port 1 MOS Pull-Up Control Register (P1PCR)

Bit	7	6	5	4	3	2	1
	P17PCR	P16PCR	P15PCR	P14PCR	P13PCR	P12PCR	P11PCR
Initial value	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1PCR is an 8-bit readable/writable register that controls the port 1 built-in MOS inpu on a bit-by-bit basis.

In modes 2 and 3, the MOS input pull-up is turned on when a P1PCR bit is set to 1 wh corresponding P1DDR bit is cleared to 0 (input port setting).

P1PCR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri software standby mode.

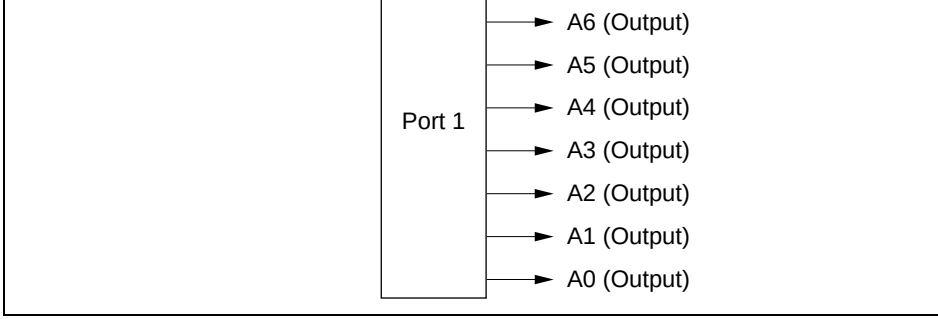


Figure 8.2 Port 1 Pin Functions (Mode 1)

Modes 2 and 3 (EXPE = 1)

In modes 2 and 3 (when EXPE = 1), port 1 pins function as address outputs, PWM outp
input ports, and input or output can be specified on a bit-by-bit basis. When a bit in P1I
to 1, the corresponding pin functions as an address output or PWM output, and when cl
as an input port.

The port 1 pin functions are shown in figure 8.3.

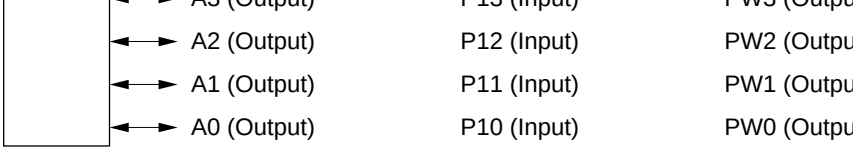


Figure 8.3 Port 1 Pin Functions (Modes 2 and 3 (EXPE = 1))

Modes 2 and 3 (EXPE = 0)

In modes 2 and 3 (when EXPE = 0), port 1 pins function as PWM outputs or I/O ports or output can be specified on a bit-by-bit basis. When a bit in P1DDR is set to 1, the corresponding pin functions as a PWM output or output port, and when cleared to 0, a port.

The port 1 pin functions are shown in figure 8.4.

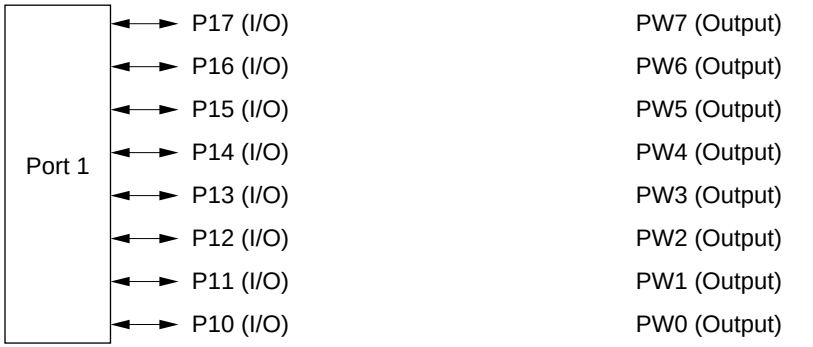


Figure 8.4 Port 1 Pin Functions (Modes 2 and 3 (EXPE = 0))

The MOS input pull-up function is in the off state after a reset and in hardware standby prior state is retained in software standby mode.

Table 8.5 summarizes the MOS input pull-up states.

Table 8.5 MOS Input Pull-Up States (Port 1)

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

Legend:
Off: MOS input pull-up is always off.
On/Off: On when P1DDR = 0 and P1PCR = 1; otherwise off.

8.3 Port 2

8.3.1 Overview

Port 2 is an 8-bit I/O port. Port 2 pins also function as address bus output pins (A15 to A PWM output pins (PW15 to PW8) (H8S/2148 Group and H8S/2147N only), and the tir connection output pin (CBLANK) (H8S/2148 Group only). Port 2 functions change acc the operating mode. Port 2 has a built-in MOS input pull-up function that can be contro software.

Figure 8.5 shows the port 2 pin configuration.

P21/A9/PW9
P20/A8/PW8

A9 (Output)
A8 (Output)

Pin functions in modes 2 and 3 (EXPE = 1)

A15 (Output)/P27 (I/O)/PW15 (Output)/CBLAN
A14 (Output)/P26 (I/O)/PW14 (Output)
A13 (Output)/P25 (I/O)/PW13 (Output)
A12 (Output)/P24 (I/O)/PW12 (Output)
A11 (Output)/P23 (Input)/PW11 (Output)
A10 (Output)/P22 (Input)/PW10 (Output)
A9 (Output)/P21 (Input)/PW9 (Output)
A8 (Output)/P20 (Input)/PW8 (Output)

Pin functions in modes 2 and 3 (EXPE = 0)

P27 (I/O)/PW15 (Output)/CBLANK (Output)
P26 (I/O)/PW14 (Output)
P25 (I/O)/PW13 (Output)
P24 (I/O)/PW12 (Output)
P23 (I/O)/PW11 (Output)
P22 (I/O)/PW10 (Output)
P21 (I/O)/PW9 (Output)
P20 (I/O)/PW8 (Output)

Figure 8.5 Port 2 Pin Functions

Port 2 MOS pull-up control register	P2PCR	R/W	H'00	H'FF

Note: * Lower 16 bits of the address.

Port 2 Data Direction Register (P2DDR)

Bit	7	6	5	4	3	2	1	
	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

P2DDR is an 8-bit write-only register, the individual bits of which specify input or outp pins of port 2. P2DDR cannot be read; if it is, an undefined value will be returned.

P2DDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri software standby mode. The address output pins maintain their output state in a transiti software standby mode.

- Mode 1

  The corresponding port 2 pins are address outputs, regardless of the P2DDR setting

  In hardware standby mode, the address outputs go to the high-impedance state.

- Modes 2 and 3 (EXPE = 1)

  The corresponding port 2 pins are address outputs or PWM outputs when P2DDR b to 1, and input ports when cleared to 0. P27 to P24 are switched from address outpu ports by setting the IOSE bit to 1.

  P27 can be used as an on-chip supporting module output pin regardless of the P27D but to ensure normal access to external space, P27 should not be set as an on-chip s module output pin when port 2 pins are used as address output pins.

	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR
Initial value	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2DR is an 8-bit readable/writable register that stores output data for the port 2 pins (
If a port 2 read is performed while P2DDR bits are set to 1, the P2DR values are read
regardless of the actual pin states. If a port 2 read is performed while P2DDR bits are
the pin states are read.

P2DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prio
software standby mode.

Port 2 MOS Pull-Up Control Register (P2PCR)

Bit	7	6	5	4	3	2	1
	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR
Initial value	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2PCR is an 8-bit readable/writable register that controls the port 2 built-in MOS inpu
on a bit-by-bit basis.

In modes 2 and 3, the MOS input pull-up is turned on when a P2PCR bit is set to 1 wh
corresponding P2DDR bit is cleared to 0 (input port setting).

P2PCR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri
software standby mode.

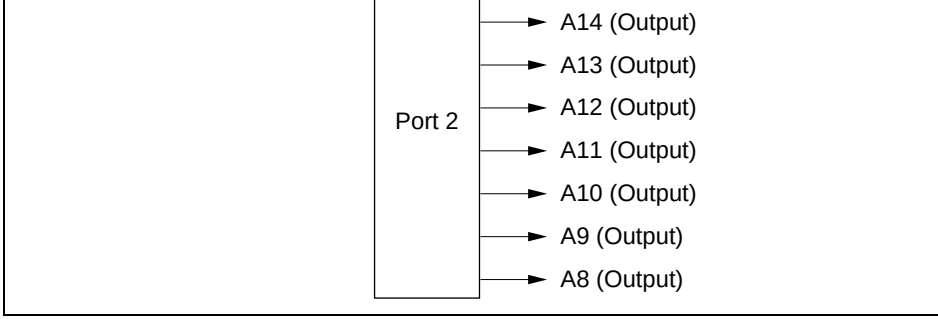


Figure 8.6 Port 2 Pin Functions (Mode 1)

Modes 2 and 3 (EXPE = 1)

In modes 2 and 3 (when EXPE = 1), port 2 pins function as address outputs, PWM outp ports, and input or output can be specified on a bit-by-bit basis. When a bit in P2DDR i the corresponding pin functions as an address output or PWM output, and when cleared input port. P27 to P24 are switched from address outputs to output ports by setting the 1. P27 can be used as an on-chip supporting module output pin regardless of the P27DD but to ensure normal access to external space, P27 should not be set as an on-chip supp module output pin when port 2 pins are used as address output pins.

The port 2 pin functions are shown in figure 8.7.

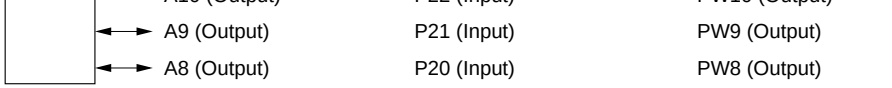


Figure 8.7 Port 2 Pin Functions (Modes 2 and 3 (EXPE = 1))

Modes 2 and 3 (EXPE = 0)

In modes 2 and 3 (when EXPE = 0), port 2 pins function as PWM outputs (P27 can al
as the timer connection output (CBLANK)) or I/O ports, and input or output can be sp
bit-by-bit basis. When a bit in P2DDR is set to 1, the corresponding pin functions as a
output or output port, and when cleared to 0, as an input port. P27 can be used as an o
supporting module output pin regardless of the P27DDR setting.

The port 2 pin functions are shown in figure 8.8.

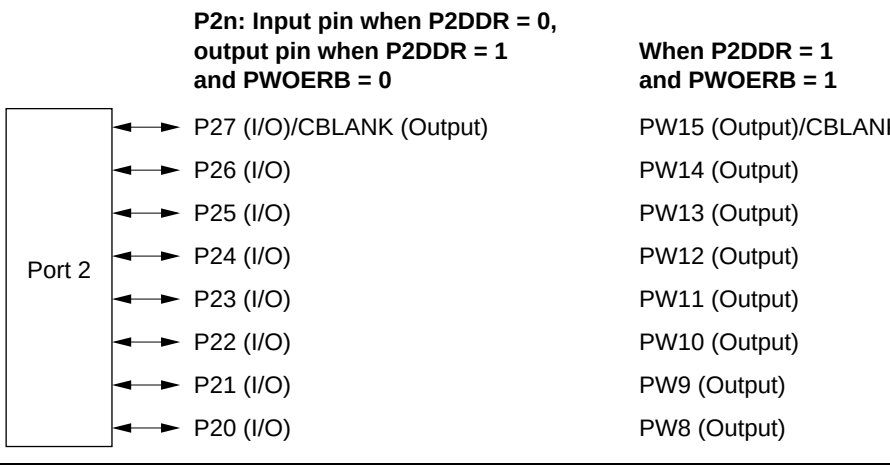


Figure 8.8 Port 2 Pin Functions (Modes 2 and 3 (EXPE = 0))

The MOS input pull-up function is in the off state after a reset and in hardware standby prior state is retained in software standby mode.

Table 8.7 summarizes the MOS input pull-up states.

Table 8.7 MOS Input Pull-Up States (Port 2)

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

Legend:

Off: MOS input pull-up is always off.

On/Off: On when P2DDR = 0 and P2PCR = 1; otherwise off.

Figure 8.9 shows the port 3 pin configuration.

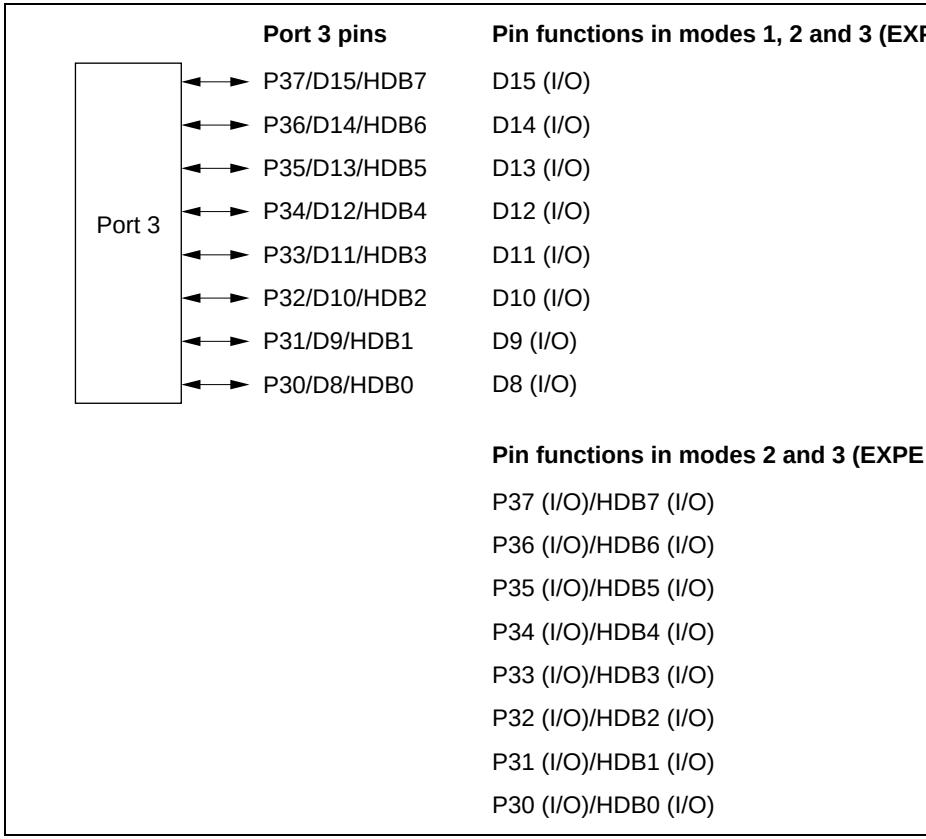


Figure 8.9 Port 3 Pin Functions

Port 3 MOS pull-up control register	P3PCR	R/W	H'00	H'FF

Note: * Lower 16 bits of the address.

Port 3 Data Direction Register (P3DDR)

Bit	7	6	5	4	3	2	1	
	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

P3DDR is an 8-bit write-only register, the individual bits of which specify input or outp pins of port 3. P3DDR cannot be read; if it is, an undefined value will be returned.

P3DDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri software standby mode.

- Modes 1, 2, and 3 (EXPE = 1)

  The input/output direction specified by P3DDR is ignored, and pins automatically f data I/O pins.

  After a reset, and in hardware standby mode or software standby mode, the data I/C the high-impedance state.

- Modes 2 and 3 (EXPE = 0)

  The corresponding port 3 pins are output ports when P3DDR bits are set to 1, and i when cleared to 0.

If a port 3 read is performed while P3DDR bits are set to 1, the P3DR values are read regardless of the actual pin states. If a port 3 read is performed while P3DDR bits are the pin states are read.

P3DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prio software standby mode.

Port 3 MOS Pull-Up Control Register (P3PCR)

Bit	7	6	5	4	3	2	1
	P37PCR	P36PCR	P35PCR	P34PCR	P33PCR	P32PCR	P31PCR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P3PCR is an 8-bit readable/writable register that controls the port 3 built-in MOS inpu on a bit-by-bit basis.

In modes 2 and 3 (when EXPE = 0), the MOS input pull-up is turned on when a P3PC to 1 while the corresponding P3DDR bit is cleared to 0 (input port setting).

P3PCR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri software standby mode.

The MOS input pull-up function cannot be used in slave mode (when the host interfac enabled).

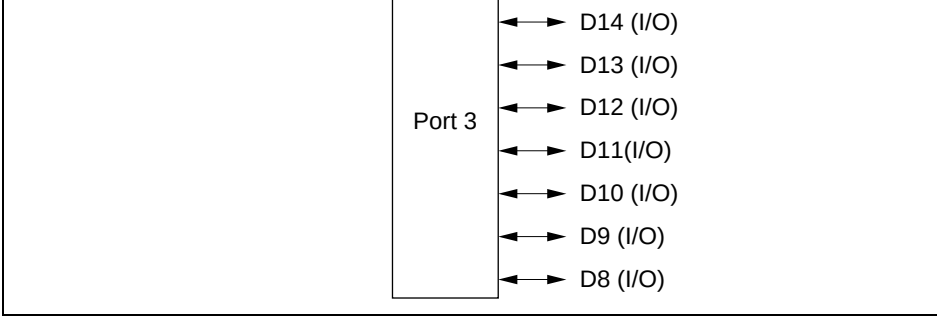


Figure 8.10 Port 3 Pin Functions (Modes 1, 2, and 3 (EXPE = 1))

Modes 2 and 3 (EXPE = 0)

In modes 2 and 3 (when EXPE = 0), port 3 functions as host interface data bus I/O pins HDB0) or as I/O ports. When the HI12E bit is set to 1 in SYSCR2 and a transition is m slave mode, port 3 functions as the host interface data bus. In slave mode, P3DR and P should be cleared to H'00. When the HI12E bit is cleared to 0, port 3 functions as an I/ input or output can be specified on a bit-by-bit basis. When a bit in P3DDR is set to 1, corresponding pin functions as an output port, and when cleared to 0, as an input port.

The port 3 pin functions are shown in figure 8.11.

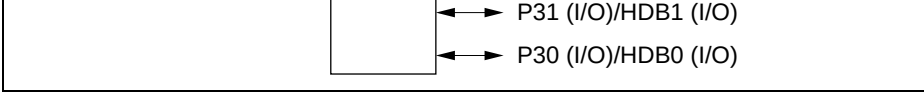


Figure 8.11 Port 3 Pin Functions (Modes 2 and 3 (EXPE = 0))

8.4.4 MOS Input Pull-Up Function

Port 3 has a built-in MOS input pull-up function that can be controlled by software. T input pull-up function can be used in modes 2 and 3 (when EXPE = 0), and can be spe or off on a bit-by-bit basis.

When a P3DDR bit is cleared to 0 in mode 2 or 3 (when EXPE = 0), setting the corres P3PCR bit to 1 turns on the MOS input pull-up for that pin.

The MOS input pull-up function is in the off state after a reset and in hardware standb prior state is retained in software standby mode.

Table 8.9 summarizes the MOS input pull-up states.

Table 8.9 MOS Input Pull-Up States (Port 3)

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

Legend:

Off: MOS input pull-up is always off.

On/Off: On when P3DDR = 0 and P3PCR = 1; otherwise off.

(HIRQ12, HIRQ1, HIRQ11) (H8S/2148 Group and H8S/2147N only), and the IIC1 I/C
(SDA1) (option in H8S/2148 Group and H8S/2147N only). Port 4 pin functions are the
operating modes.

Figure 8.12 shows the port 4 pin configuration.

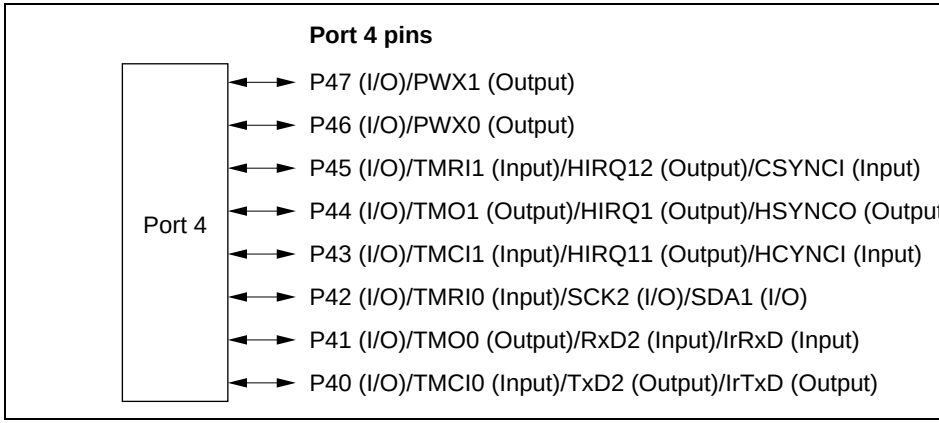


Figure 8.12 Port 4 Pin Functions

8.5.2 Register Configuration

Table 8.10 shows the port 4 register configuration.

Table 8.10 Port 4 Registers

Name	Abbreviation	R/W	Initial Value	Addr
Port 4 data direction register	P4DDR	W	H'00	H'FF
Port 4 data register	P4DR	R/W	H'00	H'FF

Note: * Lower 16 bits of the address.

P4DDR is an 8-bit write-only register, the individual bits of which specify input or ou
pins of port 4. P4DDR cannot be read; if it is, an undefined value will be returned.

When a bit in P4DDR is set to 1, the corresponding pin functions as an output port, an
cleared to 0, as an input port.

P4DDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pr
software standby mode. As 14-bit PWM and SCI2 are initialized in software standby
pin states are determined by the TMR0, TMR1, HIF, IIC1, P4DDR, and P4DR specifi

Port 4 Data Register (P4DR)

Bit	7	6	5	4	3	2	1
	P47DR	P46DR	P45DR	P44DR	P43DR	P42DR	P41DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P4DR is an 8-bit readable/writable register that stores output data for the port 4 pins (
If a port 4 read is performed while P4DDR bits are set to 1, the P4DR values are read
regardless of the actual pin states. If a port 4 read is performed while P4DDR bits are
the pin states are read.

P4DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prio
software standby mode.

8.5.3 Pin Functions

Port 4 pins also function as 14-bit PWM output pins (PWX1, PWX0), 8-bit timer 0 an
TMR1) I/O pins (TMCI0, TMRI0, TMO0, TMCI1, TMRI1, TMO1), timer connection
(CSYNCI, HSYNCI, HSYNCO), SCI2 I/O pins (TxD2, RxD2, SCK2), IrDA interfac

P47DDR	0	1	-
Pin function	P47 input pin	P47 output pin	PWX1 c

P46/PWX0

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

OEA	0		
P46DDR	0	1	-
Pin function	P46 input pin	P46 output pin	PWX0 c

P45/TMRI1/ HIRQ12/CSYNCI

The pin function is switched as shown below according to the comb the operating mode and bit P45DDR.

P45DDR	0	1	
Operating mode	—	Not slave mode	Slave
Pin function	P45 input pin	P45 output pin	HIRQ12
	TMRI1 input pin, CSYNCI input pin		

When bits CCLR1 and CCLR0 in TCR1 of TMR1 are set to 1, this p as the TMRI1 input pin. It can also be used as the CSYNCI input pi

P44/TMO1/ HIRQ1/HSYNCO

The pin function is switched as shown below according to the comb the operating mode, bits OS3 to OS0 in TCSR of TMR1, bit HOE in of the timer connection function, and bit P44DDR.

HOE	0				
OS3 to OS0	All 0			Not all 0	
P44DDR	0	1		—	
Operating mode	—	Not slave mode	Slave mode	—	
Pin function	P44 input pin	P44 output pin	HIRQ1 output pin	TMO1 output pin	

When an external clock is selected with bits CKS2 to CKS0 in TCR this pin is used as the TMCI1 input pin. It can also be used as the input pin.

P42/TMRI0/ SCK2/SDA1

The pin function is switched as shown below according to the com bit ICE in ICCR of IIC1, bits CKE1 and CKE0 in SCR of SCI2, bit C of SCI2, and bit P42DDR.

ICE	0				
CKE1	0				1
C/$\overline{A}$	0			1	—
CKE0	0		1	—	—
P42DDR	0	1	—	—	—
Pin function	P42 input pin	P42 output pin	SCK2 output pin	SCK2 output pin	SCK2 input pin
	TMRI0 input pin				

When this pin is used as the SDA1 I/O pin, bits CKE1 and CKE0 in SCI2 and bit C/$\overline{A}$ in SMR of SCI2 must all be cleared to 0. SDA1 is only output, and has direct bus drive capability.

When bits CCLR1 and CCLR0 in TCR0 of TMR0 are set to 1, this as the TMRI0 input pin.

When this pin is used as the TMO0 output pin, bit RE in SCR of SC cleared to 0.

P40/TMCI0/TxD2/ IrTxD

The pin function is switched as shown below according to the comb bit TE in SCR of SCI2 and bit P40DDR.

TE	0		
P40DDR	0	1	-
Pin function	P40 input pin	P40 output pin	TxD2 outp
	TMCI0 input pin		

When an external clock is selected with bits CKS2 to CKS0 in TCR(this pin is used as the TMCI0 input pin.

Figure 8.13 shows the port 5 pin configuration.

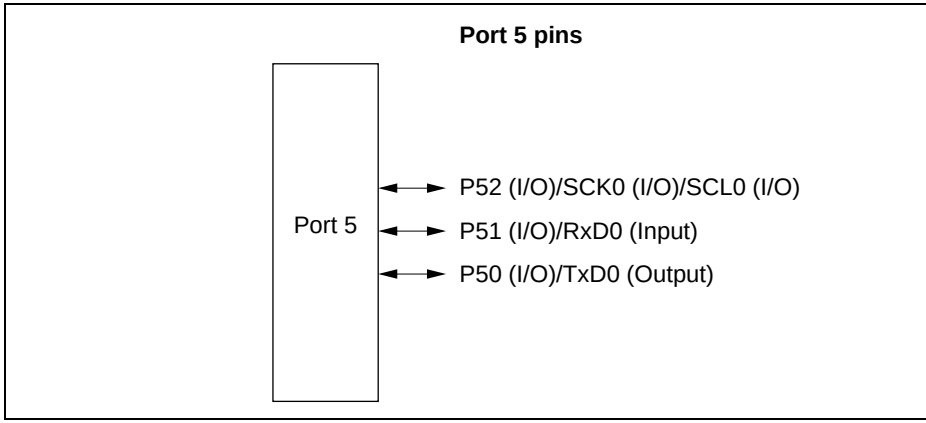


Figure 8.13 Port 5 Pin Functions

8.6.2 Register Configuration

Table 8.12 shows the port 5 register configuration.

Table 8.12 Port 5 Registers

Name	Abbreviation	R/W	Initial Value	Add
Port 5 data direction register	P5DDR	W	H'F8	H'FF
Port 5 data register	P5DR	R/W	H'F8	H'FF

Note: * Lower 16 bits of the address.

pins of port 5. P5DDR cannot be read; if it is, an undefined value will be returned. Bits
reserved.

Setting a P5DDR bit to 1 makes the corresponding port 5 pin an output pin, while clear
to 0 makes the pin an input pin.

P5DDR is initialized to H'F8 by a reset and in hardware standby mode. It retains its pri
software standby mode. As SCI0 is initialized, the pin states are determined by the IIC0
P5DDR, and P5DR specifications.

Port 5 Data Register (P5DR)

Bit	7	6	5	4	3	2	1	
	—	—	—	—	—	P52DR	P51DR	
Initial value	1	1	1	1	1	0	0	
Read/Write	—	—	—	—	—	R/W	R/W	

P5DR is an 8-bit readable/writable register that stores output data for the port 5 pins (P
If a port 5 read is performed while P5DDR bits are set to 1, the P5DR values are read d
regardless of the actual pin states. If a port 5 read is performed while P5DDR bits are c
the pin states are read.

Bits 7 to 3 are reserved; they cannot be modified and are always read as 1.

P5DR is initialized to H'F8 by a reset and in hardware standby mode. It retains its prior
software standby mode.

bits CKE1 and CKE0 in SCR of SCI0, bit C/$\overline{A}$ in SMR of SCI0, bit I
of IIC0, and bit P52DDR.

ICE	0				
CKE1	0				1
C/$\overline{A}$	0			1	—
CKE0	0		1	—	—
P52DDR	0	1	—	—	—
Pin function	P52 input pin	P52 output pin	SCK0 output pin	SCK0 output pin	SCK0 input pi

When this pin is used as the SCL0 I/O pin, bits CKE1 and CKE0 in
SCI0 and bit C/$\overline{A}$ in SMR of SCI0 must all be cleared to 0.

SCL0 is an NMOS open-drain output, and has direct bus drive cap
In the H8S/2148 Group and H8S/2147N, when set as the P52 outp
SCK0 output pin, this pin is an NMOS push-pull output.

P51/RxD0

The pin function is switched as shown below according to the com
bit RE in SCR of SCI0 and bit P51DDR.

RE	0		
P51DDR	0	1	
Pin function	P51 input pin	P51 output pin	RxD0

P50/TxD0

The pin function is switched as shown below according to the com
bit TE in SCR of SCI0 and bit P50DDR.

TE	0		
P50DDR	0	1	
Pin function	P50 input pin	P50 output pin	TxD0

pins (KIN7 to KIN0), expansion A/D converter input pins (CIN7 to CIN0), and externa
input pins ($\overline{\text{IRQ7}}$, $\overline{\text{IRQ6}}$). In the H8S/2148 Group and H8S/2147N, the port 6 input leve
switched in four stages. Port 6 pin functions are the same in all operating modes.

Figure 8.14 shows the port 6 pin configuration.

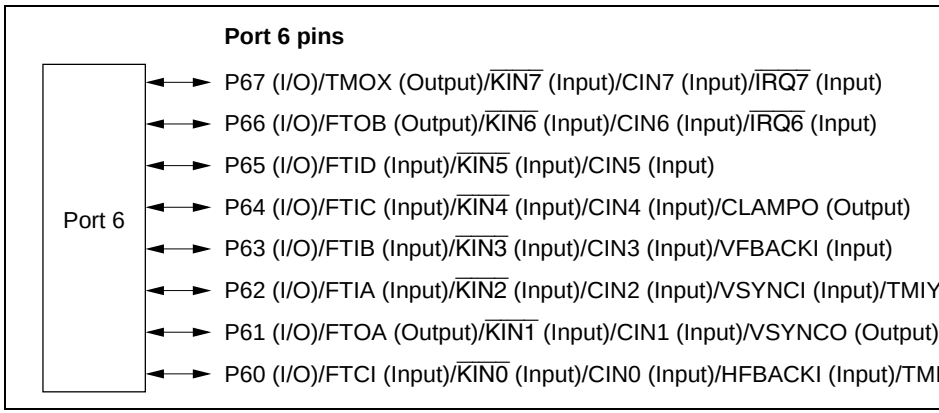


Figure 8.14 Port 6 Pin Functions

Port 6 MOS pull-up control register	KMPCR	R/W	H'00	H'
System control register 2	SYSCR2	R/W	H'00	H'

Notes: 1. Lower 16 bits of the address.
2. KMPCR has the same address as TICRR/TCORAY of TMRX/TMRY. To se
KMPCR, set the HIE bit to 1 in SYSCR and set the MSTP2 bit to 0 in MSTP

Port 6 Data Direction Register (P6DDR)

Bit	7	6	5	4	3	2	1
	P67DDR	P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

P6DDR is an 8-bit write-only register, the individual bits of which specify input or ou
pins of port 6. P6DDR cannot be read; if it is, an undefined value will be returned.

Setting a P6DDR bit to 1 makes the corresponding port 6 pin an output pin, while clea
to 0 makes the pin an input pin.

P6DDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pr
software standby mode.

regardless of the actual pin states. If a port 6 read is performed while P6DDR bits are cl
the pin states are read.

P6DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prior
software standby mode.

Port 6 MOS Pull-Up Control Register (KMPCR)

Bit	7	6	5	4	3	2	1	
	KM7PCR	KM6PCR	KM5PCR	KM4PCR	KM3PCR	KM2PCR	KM1PCR	K
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

KMPCR is an 8-bit readable/writable register that controls the port 6 built-in MOS inpu
on a bit-by-bit basis.

The MOS input pull-up is turned on when a KMPCR bit is set to 1 while the correspond
P6DDR bit is cleared to 0 (input port setting).

KMPCR is initialized to H'00 by a reset and in hardware standby mode. It retains its pr
software standby mode.

operation of host interface functions.

Only bits 7, 6, and 5 are described here. See section 18.2.2, System Control Register 2 for information on bits 4 to 0.

SYSCR2 is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 and 6—Key Wakeup Level 1 and 0 (KWUL1, KWUL0): The port 6 input le can be changed by software, using these bits. The setting of these bits also changes the of the pin functions multiplexed with port 6.

Bit 7	Bit 6		
KWUL1	**KWUL0**	**Description**	
0	0	Standard input level is selected as port 6 input level	(
	1	Input level 1 is selected as port 6 input level	
1	0	Input level 2 is selected as port 6 input level	
	1	Input level 3 is selected as port 6 input level	

Bit 5—Port 6 Input Pull-Up Extra (P6PUE): Controls and selects the current specif the port 6 MOS input pull-up function connected by means of KMPCR settings.

Bit 5		
P6PUE	**Description**	
0	Standard current specification is selected for port 6 MOS input pull-up f	(
1	Current-limit specification is selected for port 6 MOS input pull-up functi	

Table 8.15 Port 6 Pin Functions

Pin	Selection Method and Pin Functions
P67/TMOX/$\overline{\text{IRQ7}}$/ $\overline{\text{KIN7}}$/CIN7	The pin function is switched as shown below according to the comb bits OS3 to OS0 in TCSR of TMRX and bit P67DDR.

OS3 to OS0	All 0		Not
P67DDR	0	1	-
Pin function	P67 input pin	P67 output pin	TMOX c
	$\overline{\text{IRQ7}}$ input pin, $\overline{\text{KIN7}}$ input pin, CIN7 input p		

This pin is used as the $\overline{\text{IRQ7}}$ input pin when bit IRQ7E is set to 1 in

It can always be used as the $\overline{\text{KIN7}}$ or CIN7 input pin.

P66/FTOB/$\overline{\text{IRQ6}}$/ $\overline{\text{KIN6}}$/CIN6	The pin function is switched as shown below according to the comb bit OEB in TOCR of the FRT and bit P66DDR.

OEB	0		
P66DDR	0	1	-
Pin function	P66 input pin	P66 output pin	FTOB c
	$\overline{\text{IRQ6}}$ input pin, $\overline{\text{KIN6}}$ input pin, CIN6 input p		

This pin is used as the $\overline{\text{IRQ6}}$ input pin when bit IRQ6E is set to 1 in

It can always be used as the $\overline{\text{KIN6}}$ or CIN6 input pin.

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

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

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

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

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

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

This pin can always be used as the FTIB, $\overline{\text{KIN3}}$, CIN3, or VFBACK

P62/FTIA/TMIY/$\overline{\text{KIN2}}$/CIN2/VSYNCI

P62DDR	0	1
Pin function	P62 input pin	P62 output
	FTIA input pin, VSYNCI input pin, TMIY inpu $\overline{\text{KIN2}}$ input pin, CIN2 input pin	

This pin can always be used as the FTIA, TMIY, $\overline{\text{KIN2}}$, CIN2, or VS pin.

P61/FTOA/$\overline{\text{KIN1}}$/CIN1/VSYNCO

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

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

When this pin is used as the VSYNCO pin, bit OEA in TOCR of the be cleared to 0.

This pin can always be used as the $\overline{\text{KIN1}}$ or CIN1 input pin.

8.7.4 MOS Input Pull-Up Function

Port 6 has a built-in MOS input pull-up function that can be controlled by software. Th
input pull-up function can be used in any operating mode, and can be specified as on o
bit-by-bit basis.

When a P6DDR bit is cleared to 0, setting the corresponding KMPCR bit to 1 turns on
input pull-up for that pin. The MOS input pull-up current specification can be changed
of the P6PUE bit. When a pin is designated as an on-chip supporting module output pi
input pull-up is always off.

The MOS input pull-up function is in the off state after a reset and in hardware standby
prior state is retained in software standby mode.

Table 8.16 summarizes the MOS input pull-up states.

Table 8.16 MOS Input Pull-Up States (Port 6)

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

Legend:

Off: MOS input pull-up is always off.

On/Off: On when P6DDR = 0 and KMPCR = 1; otherwise off.

Figure 8.15 shows the port 7 pin configuration.

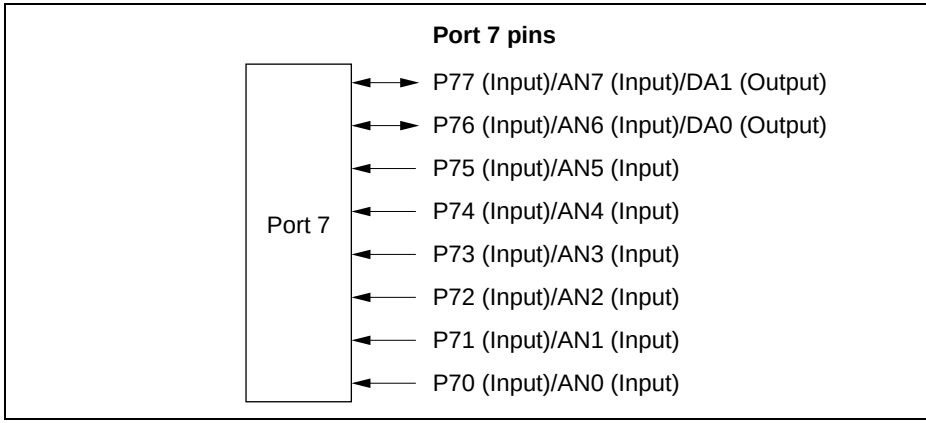


Figure 8.15 Port 7 Pin Functions

8.8.2 Register Configuration

Table 8.17 shows the port 7 register configuration. Port 7 is an input-only port, and do
a data direction register or data register.

Table 8.17 Port 7 Registers

Name	Abbreviation	R/W	Initial Value	Add
Port 7 input data register	P7PIN	R	Undefined	H'FI

Notes: 1. Lower 16 bits of the address.
2. P7PIN has the same address as PBDDR.

Note: * Determined by the state of pins P77 to P70.

When a P7PIN read is performed, the pin states are always read.

P7PIN has the same address as PBDDR; if a write is performed, data will be written int
and the port B setting will be changed.

8.8.3 Pin Functions

Port 7 pins also function as the A/D converter analog input pins (AN0 to AN7) and D/A
analog output pins (DA0, DA1).

port 8 pin configuration.

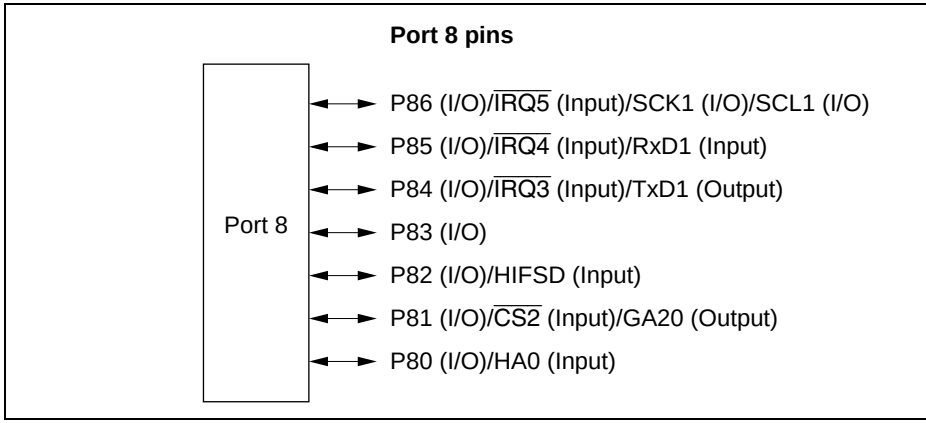


Figure 8.16 Port 8 Pin Functions

8.9.2 Register Configuration

Table 8.18 summarizes the port 8 registers.

Table 8.18 Port 8 Registers

Name	Abbreviation	R/W	Initial Value	Add
Port 8 data direction register	P8DDR	W	H'80	H'FF
Port 8 data register	P8DR	R/W	H'80	H'FF

Notes: 1. Lower 16 bits of the address.
2. P8DDR has the same address as PBPIN.

pins of port 8. P8DDR has the same address as PBPIN, and if read, the port B state will returned.

Setting a P8DDR bit to 1 makes the corresponding port 8 pin an output pin, while clear to 0 makes the pin an input pin.

P8DDR is initialized to H'80 by a reset and in hardware standby mode. It retains its pri software standby mode.

Port 8 Data Register (P8DR)

Bit	7	6	5	4	3	2	1
	—	P86DR	P85DR	P84DR	P83DR	P82DR	P81DR
Initial value	1	0	0	0	0	0	0
Read/Write	—	R/W	R/W	R/W	R/W	R/W	R/W

P8DR is a 7-bit readable/writable register that stores output data for the port 8 pins (P8 a port 8 read is performed while P8DDR bits are set to 1, the P8DR values are read dire regardless of the actual pin states. If a port 8 read is performed while P8DDR bits are c the pin states are read.

P8DR is initialized to H'80 by a reset and in hardware standby mode. It retains its prior software standby mode.

8.9.3 Pin Functions

Port 8 pins also function as SCI1 I/O pins (TxD1, RxD1, SCK1), the IIC1 I/O pin (SCL I/O pins ($\overline{CS2}$, GA20, HA0, HIFSD), and external interrupt input pins ($\overline{IRQ5}$ to $\overline{IRQ3}$). pin functions are shown in table 8.19.

C/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				

When the IRQ5E bit in IER is set to 1, this pin is used as the $\overline{\text{IRQ5}}$
When this pin is used as the SCL1 I/O pin, bits CKE1 and CKE0 in SCI1 and bit C/$\overline{\text{A}}$ in SMR of SCI1 must all be cleared to 0. SCL1 is only output, and has direct bus drive capability.

P85/$\overline{\text{IRQ4}}$/RxD1 The pin function is switched as shown below according to the com bit RE in SCR of SCI1 and bit P85DDR.

RE	0		
P85DDR	0	1	
Pin function	P85 input pin	P85 output pin	RxD1
	$\overline{\text{IRQ4}}$ input pin		

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

P84/$\overline{\text{IRQ3}}$/TxD1 The pin function is switched as shown below according to the com bit TE in SCR of SCI1 and bit P84DDR.

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

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

mode					
SDE	—		0		
P82DDR	0	1	0	1	
Pin function	P82 input pin	P82 output pin	P82 input pin	P82 output pin	

P81/GA20/$\overline{CS2}$ The pin function is switched as shown below according to the comb operating mode, bit CS2E in SYSCR, bit FGA20E in HICR of the HI P81DDR.

Operating mode	Not slave mode		Slave mode			
FGA20E	—		0			
CS2E	—		0		1	
P81DDR	0	1	0	1	—	0
Pin function	P81 input pin	P81 output pin	P81 input pin	P81 output pin	$\overline{CS2}$ input pin	P81 input pin

This pin should be used as the GA20 output pin or $\overline{CS2}$ input pin on 2 or 3 (EXPE = 0).

P80/HA0 The pin function is switched as shown below according to the comb operating mode and bit P80DDR.

Operating mode	Not slave mode		Slave
P80DDR	0	1	-
Pin function	P80 input pin	P80 output pin	HA0 in

pins (AS/IOS, RD, HWR, LWR, WAIT), and the system clock (φ) output pin. In H8S
and H8S/2147N, P97 is an NMOS push-pull output. SDA0 is an NMOS open-drain ou
has direct bus drive capability.

Figure 8.17 shows the port 9 pin configuration.

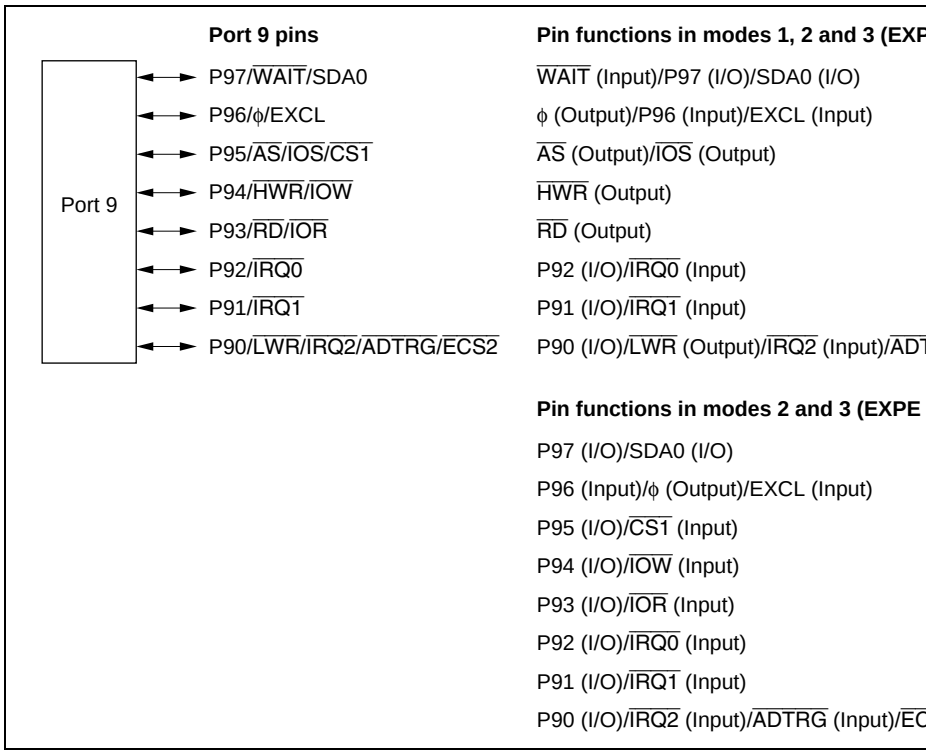


Figure 8.17 Port 9 Pin Functions

Notes: 1. Lower 16 bits of the address.
2. Initial value depends on the mode.

Port 9 Data Direction Register (P9DDR)

Bit	7	6	5	4	3	2	1	
	P97DDR	P96DDR	P95DDR	P94DDR	P93DDR	P92DDR	P91DDR	
Mode 1								
Initial value	0	1	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	
Modes 2 and 3								
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

P9DDR is an 8-bit write-only register, the individual bits of which specify input or outp
pins of port 9. P9DDR cannot be read; if it is, an undefined value will be returned.

P9DDR is initialized to H'40 (mode 1) or H'00 (modes 2 and 3) by a reset and in hardw
mode. It retains its prior state in software standby mode.

- Modes 1, 2 and 3 (EXPE = 1)

  Pin P97 functions as a bus control input ($\overline{\text{WAIT}}$), the IIC0 I/O pin (SDA0), or an I/
  according to the wait mode setting. When P97 functions as an I/O port, it becomes a
  port when P97DDR is set to 1, and an input port when P97DDR is cleared to 0.

  Pin P96 functions as the ϕ output pin when P96DDR is set to 1, and as the subclock
  (EXCL) or an input port when P96DDR is cleared to 0.

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

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

Port 9 Data Register (P9DR)

Bit	7	6	5	4	3	2	1
	P97DR	P96DR	P95DR	P94DR	P93DR	P92DR	P91DR
Initial value	0	—*	0	0	0	0	0
Read/Write	R/W	R	R/W	R/W	R/W	R/W	R/W

Note: * Determined by the state of pin P96.

P9DR is an 8-bit readable/writable register that stores output data for the port 9 pins (P
With the exception of P96, if a port 9 read is performed while P9DDR bits are set to 1
values are read directly, regardless of the actual pin states. If a port 9 read is performe
P9DDR bits are cleared to 0, the pin states are read.

P9DR is initialized to H'00 by a reset and in hardware standby mode. It retains its prio
software standby mode.

8.10.3 Pin Functions

Port 9 pins also function as external interrupt input pins ($\overline{\text{IRQ0}}$ to $\overline{\text{IRQ2}}$), the A/D con
input pin ($\overline{\text{ADTRG}}$), HIF input pins ($\overline{\text{ECS2}}$, $\overline{\text{CS1}}$, $\overline{\text{IOW}}$, $\overline{\text{IOR}}$), the IIC0 I/O pin (SDA0
subclock input pin (EXCL), bus control signal I/O pins ($\overline{\text{AS}}$/$\overline{\text{IOS}}$, $\overline{\text{RD}}$, $\overline{\text{HWR}}$, $\overline{\text{LWR}}$, $\overline{\text{W}}$
the system clock (ϕ) output pin. The pin functions differ between the mode 1, 2, and 3
expanded modes and the mode 2 and 3 (EXPE = 0) single-chip modes. The port 9 pin
are shown in table 8.21.

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 outpu pin

When this pin is set as the P97 output pin in the H8S/2148 Group a H8S/2147N, it is an NMOS push-pull output. SDA0 is an NMOS ope output, and has direct bus drive capability.

P96/φ/EXCL

The pin function is switched as shown below according to the comb bit EXCLE in LPWRCR and bit P96DDR.

P96DDR	0		
EXCLE	0	1	
Pin function	P96 input pin	EXCL input pin	φ outp

When this pin is used as the EXCL input pin, P96DDR should be cle

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

The pin function is switched as shown below according to the comb operating mode, bit IOSE in SYSCR, bit HI12E in SYSCR2, and bit

Operating mode	Modes 1, 2, 3 (EXPE = 1)		Modes 2, 3 (EXPE		
HI12E	—		0		
P95DDR	—		0	1	
IOSE	0	1	—	—	
Pin function	$\overline{\text{AS}}$ output pin	$\overline{\text{IOS}}$ output pin	P95 input pin	P95 output pin	

	output pin	input pin	output pin	

P93/$\overline{\text{RD}}$/$\overline{\text{IOR}}$ The pin function is switched as shown below according to the com operating mode, bit HI12E in SYSCR2, and bit P93DDR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)	Modes 2, 3 (EXPE = 0)		
HI12E	—	0		
P93DDR	—	0	1	
Pin function	$\overline{\text{RD}}$ output pin	P93 input pin	P93 output pin	$\overline{\text{IO}}$

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

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

When bit IRQ0E in IER is set to 1, this pin is used as the $\overline{\text{IRQ0}}$ inp

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

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

When bit IRQ1E in IER is set to 1, this pin is used as the $\overline{\text{IRQ1}}$ inp

FGA20E	—				
CS2E	—				
P90DDR	—	0	1	0	1
Pin function	$\overline{\text{LWR}}$ output pin	P90 input pin	P90 output pin	P90 input pin	P90 output pin
		$\overline{\text{IRQ2}}$ input pin, $\overline{\text{ADTRG}}$ input pi			

When the IRQ2E bit in IER is set to 1 in mode 1, 2, or 3 (EXPE = 1) ABW bit in WSCR set to 1, or in mode 2 and 3 (EXPE = 0), this pin the $\overline{\text{IRQ2}}$ input pin.

When TRGS1 and TRGS0 in ADCR of the A/D converter are both s pin is used as the $\overline{\text{ADTRG}}$ input pin.

modes. Figure 8.18 shows the port A pin configuration.

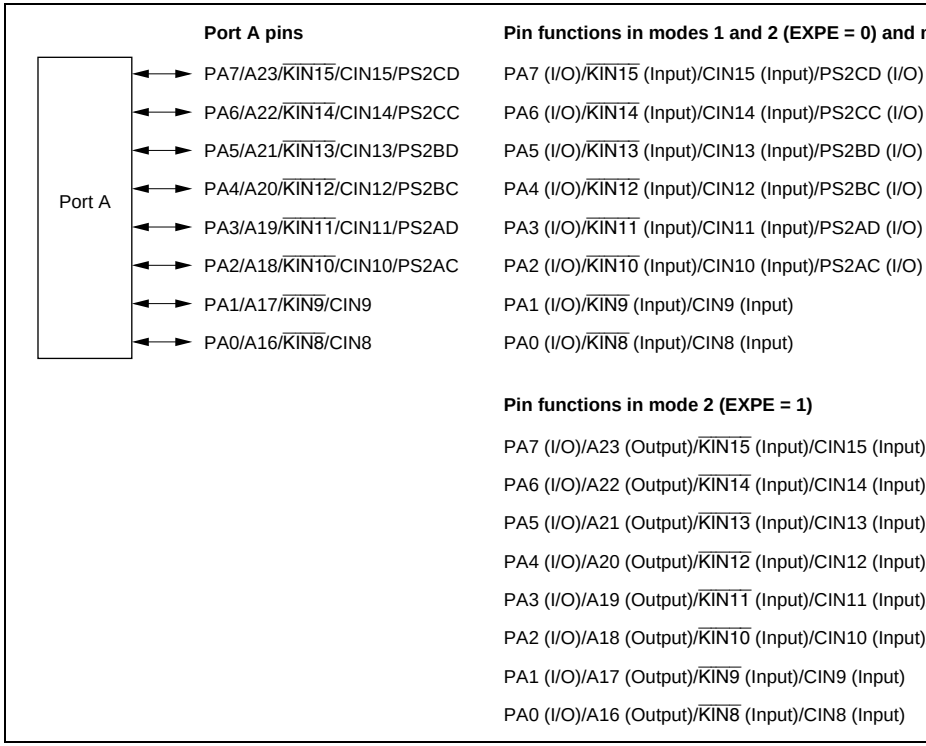


Figure 8.18 Port A Pin Functions

Port A input data register	PAPIN	R	Undefined	H'FF

Notes: 1. Lower 16 bits of the address.
2. PADDR and PAPIN have the same address.

Port A Data Direction Register (PADDR)

Bit	7	6	5	4	3	2	1	
	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

PADDR is an 8-bit write-only register, the individual bits of which specify input or out
pins of port A.

Setting a PADDR bit to 1 makes the corresponding port A pin an output pin, while clea
to 0 makes the pin an input pin.

PADDR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri
software standby mode.

Port A Output Data Register (PAODR)

Bit	7	6	5	4	3	2	1	
	PA7ODR	PA6ODR	PA5ODR	PA4ODR	PA3ODR	PA2ODR	PA1ODR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

PAODR is an 8-bit readable/writable register that stores output data for the port A pins
PA0). PAODR can always be read or written to, regardless of the contents of PADDR.

Read/Write	R	R	R	R	R	R	R

Note: * Determined by the state of pins PA7 to PA0.

Reading PAPIN always returns the pin states.

8.11.3 Pin Functions

Port A pins also function as keyboard buffer controller I/O pins (PS2AC, PS2AD, PS2
PS2BD, PS2CC, PS2CD), key-sense interrupt input pins ($\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$), expansion
converter input pins (CIN15 to CIN8), and address output pins (A23 to A16). The por
functions are shown in table 8.23.

Table 8.23 Port A Pin Functions

Pin	Selection Method and Pin Functions
PA7/A23/PS2CD/ $\overline{\text{KIN15}}$/CIN15	The pin function is switched as shown below according to the com operating mode, the KBIOE bit in KBCR2H of the keyboard buffer the IOSE bit in SYSCR, and bit PA7DDR.

Operating mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE		
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	PA outp pin
	$\overline{\text{KIN15}}$ input pin, CIN15 input pin, PS2CD inp					

When the IICS bit in STCR is set to 1, this pin functions as a bus b
pin can always be used as the PS2CD, $\overline{\text{KIN15}}$, or CIN15 input pin.

IOSE	—	—	—	—	0	1
Pin function	PA6 input pin	PA6 output pin	PS2CC output pin	PA6 input pin	A22 output pin	PA6 outpu pin
	$\overline{\text{KIN14}}$ input pin, CIN14 input pin, PS2CC inpu					

When the IICS bit in STCR is set to 1, this pin functions as a bus bu
pin can always be used as the PS2CC, $\overline{\text{KIN14}}$, or CIN14 input pin.

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

The pin function is switched as shown below according to the comb
operating mode, the KBIOE bit in KBCR1H of the keyboard buffer c
the IOSE bit in SYSCR, and bit PA5DDR.

Operating mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE =		
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 outpu pin
	$\overline{\text{KIN13}}$ input pin, CIN13 input pin, PS2BD inpu					

When the IICS bit in STCR is set to 1, this pin functions as a bus bu
pin can always be used as the PS2BD, $\overline{\text{KIN13}}$, or CIN13 input pin.

IOSE	—	—	—	—	0	1
Pin function	PA4 input pin	PA4 output pin	PS2BC output pin	PA4 input pin	A20 output pin	PA4 outp pin
	$\overline{\text{KIN12}}$ input pin, CIN12 input pin, PS2BC inp					

When the IICS bit in STCR is set to 1, this pin functions as a bus b
pin can always be used as the PS2BC, $\overline{\text{KIN12}}$, or CIN12 input pin.

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

The pin function is switched as shown below according to the com
operating mode, the KBIOE bit in KBCR0H of the keyboard buffer
the IOSE bit in SYSCR, and bit PA3DDR.

Operating mode	Modes 1, 2 (EXPE = 0), 3			Mode 2 (EXPE		
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 outp pin
	$\overline{\text{KIN11}}$ input pin, CIN11 input pin, PS2AD inp					

This pin can always be used as the PS2AD, $\overline{\text{KIN11}}$, or CIN11 inpu

IOSE	—	—	—	—	0	1
Pin function	PA2 input pin	PA2 output pin	PS2AC output pin	PA2 input pin	A18 output pin	PA2 outpu pin
	$\overline{\text{KIN10}}$ input pin, CIN10 input pin, PS2AC input					

This pin can always be used as the PS2AC, $\overline{\text{KIN10}}$, or CIN10 input

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

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

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

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

PA0/A16/$\overline{\text{KIN8}}$/ CIN8

The pin function is switched as shown below according to the comb operating mode, the IOSE bit in SYSCR, and bit PA0DDR.

Operating mode	Modes 1, 2 (EXPE = 0), 3		Mode 2 (EXPE =	
PA0DDR	0	1	0	1
IOSE	—	—	—	0
Pin function	PA0 input pin	PA0 output pin	PA0 input pin	A16 output pin
	$\overline{\text{KIN8}}$ input pin, CIN8 input pin			

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

input pull-up is always off.

The MOS input pull-up function is in the off state after a reset and in hardware standb
prior state is retained in software standby mode.

Table 8.24 summarizes the MOS input pull-up states.

Table 8.24 MOS Input Pull-Up States (Port A)

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

Legend:

Off: MOS input pull-up is always off.

On/Off: On when PADDR = 0 and PAODR = 1; otherwise off.

Figure 8.19 shows the port B pin configuration.

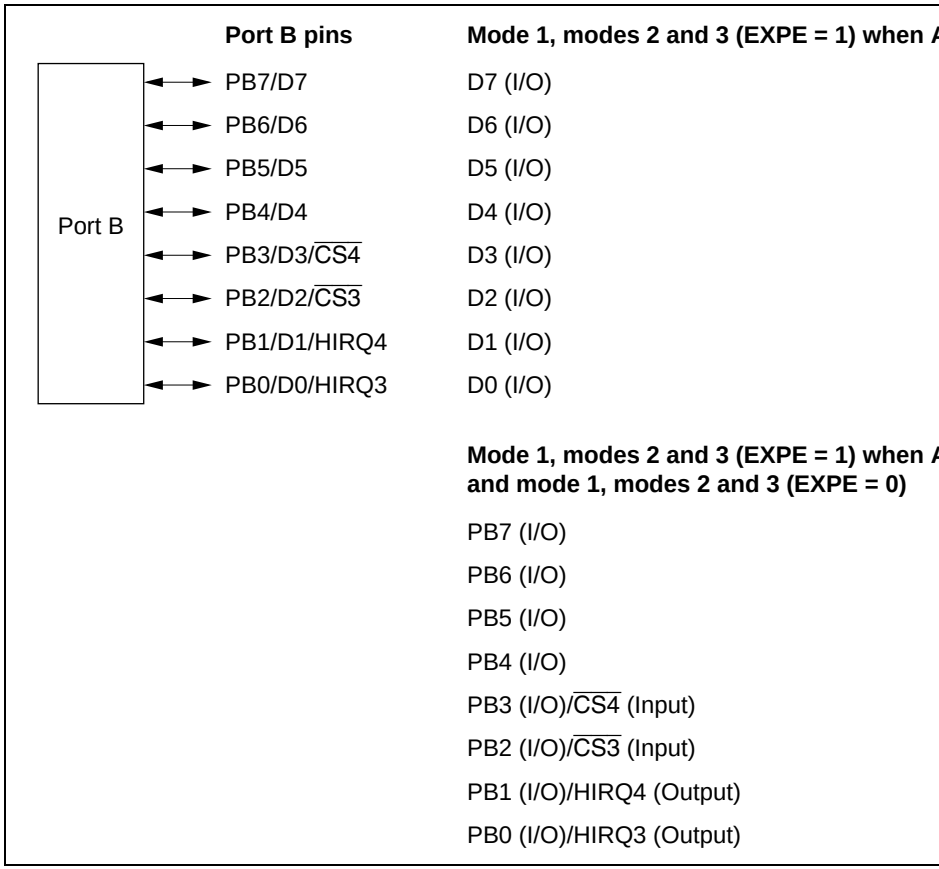


Figure 8.19 Port B Pin Functions

Port B input data register	PBPIN	R	Undefined	H'FF

Notes: 1. Lower 16 bits of the address.
2. PBDDR has the same address as P7PIN.
3. PBPIN has the same address as P8DDR.

Port B Data Direction Register (PBDDR)

Bit	7	6	5	4	3	2	1
	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

PBDDR is an 8-bit write-only register, the individual bits of which specify input or ou pins of port B. PBDDR has the same address as P7PIN, and if read, the port 7 pin stat returned.

Setting a PBDDR bit to 1 makes the corresponding port B pin an output pin, while cle to 0 makes the pin an input pin.

PBDDR is initialized to H'00 by a reset and in hardware standby mode. It retains its p software standby mode.

- Modes 1, 2 and 3 (EXPE = 1)

  When the ABW bit in WSCR is cleared to 0, port B pins automatically become da (D7 to D0), regardless of the input/output direction indicated by PBDDR. When th is 1, a port B pin becomes an output port if the corresponding PBDDR bit is set to input port if the bit is cleared to 0.

  Data I/O pins go to the high-impedance state after a reset, and in hardware standby software standby mode.

Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PBODR is an 8-bit readable/writable register that stores output data for the port B pins
PB0). PBODR can always be read or written to, regardless of the contents of PBDDR.

PBODR is initialized to H'00 by a reset and in hardware standby mode. It retains its pri
software standby mode.

Port B Input Data Register (PBPIN)

Bit	7	6	5	4	3	2	1
	PB7PIN	PB6PIN	PB5PIN	PB4PIN	PB3PIN	PB2PIN	PB1PIN
Initial value	—*	—*	—*	—*	—*	—*	—*
Read/Write	R	R	R	R	R	R	R

Note: * Determined by the state of pins PB7 to PB0.

Reading PBPIN always returns the pin states.

PBPIN has the same address as P8DDR. If a write is performed, data will be written to
and the port 8 settings will change.

PB7/D7

The pin function is switched as shown below according to the com
the operating mode, bit PB7DDR, and bit ABW in WSCR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)			Mode (EXPE
ABW	0	1		–
PB7DDR	—	0	1	0
Pin function	D7 I/O pin	PB7 input pin	PB7 output pin	PB7 input pin

PB6/D6

The pin function is switched as shown below according to the com
the operating mode, bit PB6DDR, and bit ABW in WSCR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)			Mode (EXPE
ABW	0	1		–
PB6DDR	—	0	1	0
Pin function	D6 I/O pin	PB6 input pin	PB6 output pin	PB6 input pin

PB5/D5

The pin function is switched as shown below according to the com
the operating mode, bit PB5DDR, and bit ABW in WSCR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)			Mode (EXPE
ABW	0	1		–
PB5DDR	—	0	1	0
Pin function	D5 I/O pin	PB5 input pin	PB5 output pin	PB5 input pin

	I/O pin	input pin	output pin	input pin	

PB3/D3/$\overline{CS4}$

The pin function is switched as shown below according to the comb
the operating mode, bits HI12E and CS4E in SYSCR2, bit ABW in V
bit PB3DDR.

Operating mode	Modes 1, 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

PB2/D2/$\overline{CS3}$

The pin function is switched as shown below according to the comb
the operating mode, bits HI12E and CS3E in SYSCR2, bit ABW in V
bit PB2DDR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)			Modes 2, 3 (EXPE = 0	
HI12E	—			Either cleared to 0	
CS3E	—				
ABW	0	1		—	
PB2DDR	—	0	1	0	1
Pin function	D2 I/O pin	PB2 input pin	PB2 output pin	PB2 input pin	PB2 output pin

	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 pi

PB0/D0/HIRQ3

The pin function is switched as shown below according to the com
the operating mode, bits HI12E and CS3E in SYSCR2, bit ABW in
bit PB0DDR.

Operating mode	Modes 1, 2, 3 (EXPE = 1)			Modes 2, (EXPE =	
HI12E	—			Either cleared to 0	
CS3E	—				
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 pi

The MOS input pull-up function is in the off state after a reset and in hardware standby prior state is retained in software standby mode.

Table 8.27 summarizes the MOS input pull-up states.

Table 8.27 MOS Input Pull-Up States (Port B)

Mode	Reset	Hardware Standby Mode	Software Standby Mode	In Oth Opera
1, 2, 3 (EXPE = 1) with ABW in WSCR = 0	Off	Off	Off	Off
1, 2, 3 (EXPE = 1) with ABW in WSCR = 1, and 2, 3 (EXPE = 0)			On/Off	On/Of

Legend:

Off: MOS input pull-up is always off.

On/Off: On when PBDDR = 0 and PBODR = 1; otherwise off.

outputs. Sixteen output waveforms are generated from a common time base, enabling output with a high carrier frequency to be produced using pulse division. The PWM ti has sixteen 8-bit PWM data registers (PWDRs), and an output pulse with a duty cycle 100% can be obtained as specified by PWDR and the port data register (P1DR or P2D

9.1.1 Features

The PWM timer module has the following features.

- Operable at a maximum carrier frequency of 1.25 MHz using pulse division (at 20 operation)
- Duty cycles from 0 to 100% with 1/256 resolution (100% duty realized by port ou
- Direct or inverted PWM output, and PWM output enable/disable control

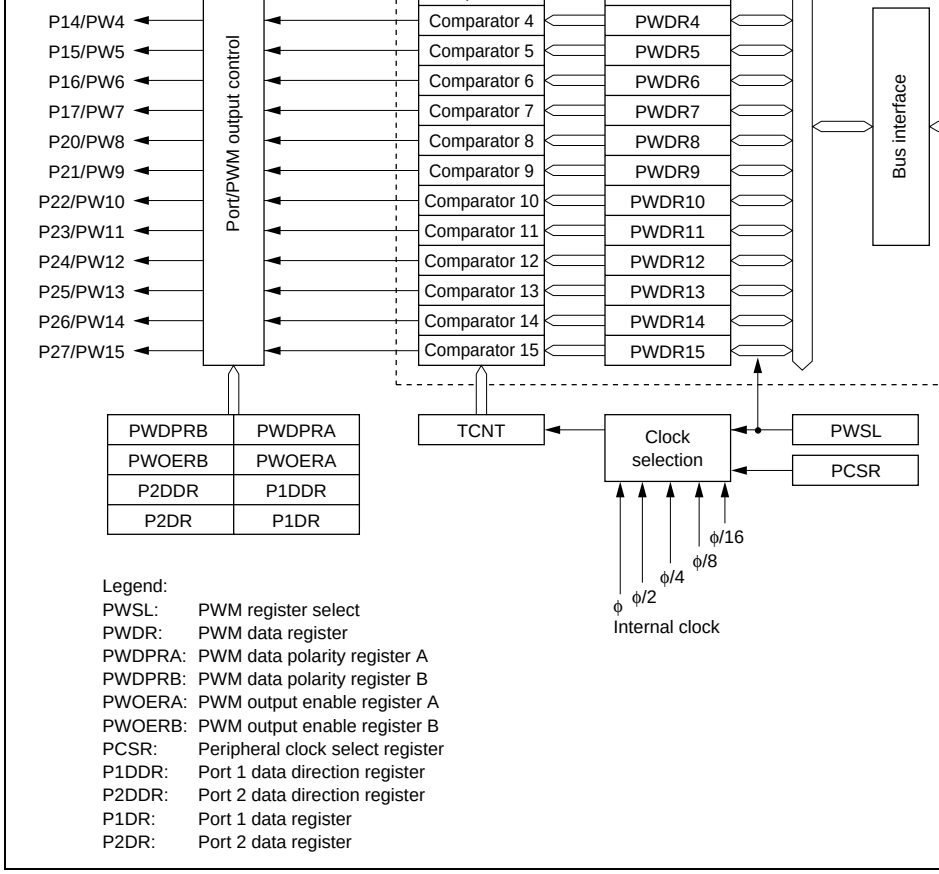


Figure 9.1 Block Diagram of PWM Timer Module

9.1.4 Register Configuration

Table 9.2 lists the registers of the PWM timer module.

Table 9.2 PWM Timer Module Registers

Name	Abbreviation	R/W	Initial Value	A
PWM register select	PWSL	R/W	H'20	H
PWM data registers 0 to 15	PWDR0 to PWDR15	R/W	H'00	H
PWM data polarity register A	PWDPRA	R/W	H'00	H
PWM data polarity register B	PWDPRB	R/W	H'00	H
PWM output enable register A	PWOERA	R/W	H'00	H
PWM output enable register B	PWOERB	R/W	H'00	H
Port 1 data direction register	P1DDR	W	H'00	H
Port 2 data direction register	P2DDR	W	H'00	H
Port 1 data register	P1DR	R/W	H'00	H
Port 2 data register	P2DR	R/W	H'00	H
Peripheral clock select register	PCSR	R/W	H'00	H
Module stop control register	MSTPCRH	R/W	H'3F	H
	MSTPCRL	R/W	H'FF	H

Note: 1. Lower 16 bits of the address.
2. Some registers in the 8-bit timer are assigned in the addresses as other reg this case, register selection is performed by the FLSHE bit in the serial time register (STCR).

Read/Write	R/W	R/W	—	—	R/W	R/W	R/W

PWSL is an 8-bit readable/writable register used to select the PWM timer input clock a
PWM data register.

PWSL is initialized to H'20 by a reset, and in the standby modes, watch mode, subactiv
subsleep mode, and module stop mode.

Bits 7 and 6—PWM Clock Enable, PWM Clock Select (PWCKE, PWCKS): These
together with bits PWCKA and PWCKB in PCSR, select the internal clock input to TC
PWM timer.

PWSL		PCSR		
Bit 7	**Bit 6**	**Bit 2**	**Bit 1**	
PWCKE	**PWCKS**	**PWCKB**	**PWCKA**	**Description**
0	—	—	—	Clock input is disabled (Ir
1	0	—	—	ϕ (system clock) is selected
	1	0	0	$\phi/2$ is selected
			1	$\phi/4$ is selected
		1	0	$\phi/8$ is selected
			1	$\phi/16$ is selected

The PWM resolution, PWM conversion period, and carrier frequency depend on the se
internal clock, and can be found from the following equations.

Resolution (minimum pulse width) = 1/internal clock frequency
PWM conversion period = resolution $\times$ 256
Carrier frequency = 16/PWM conversion period

Thus, with a 20-MHz system clock (ϕ), the resolution, PWM conversion period, and ca
frequency are as shown below.

φ/16	800 ns	204.8 μs	78.1 kHz

Bit 5—Reserved: This bit is always read as 1 and cannot be modified.

Bit 4—Reserved: This bit is always read as 0 and cannot be modified.

Bits 3 to 0—Register Select (RS3 to RS0): These bits select the PWM data register.

Bit 3	Bit 2	Bit 1	Bit 0	
RS3	**RS2**	**RS1**	**RS0**	**Register Selection**
0	0	0	0	PWDR0 selected
			1	PWDR1 selected
		1	0	PWDR2 selected
			1	PWDR3 selected
	1	0	0	PWDR4 selected
			1	PWDR5 selected
		1	0	PWDR6 selected
			1	PWDR7 selected
1	0	0	0	PWDR8 selected
			1	PWDR9 selected
		1	0	PWDR10 selected
			1	PWDR11 selected
	1	0	0	PWDR12 selected
			1	PWDR13 selected
		1	0	PWDR14 selected
			1	PWDR15 selected

be output, and the number of additional pulses. The value set in PWDR corresponds to ratio in the conversion period. The upper 4 bits specify the duty cycle of the basic pulse 15/16 with a resolution of 1/16. The lower 4 bits specify how many extra pulses are to within the conversion period comprising 16 basic pulses. Thus, a specification of 0/256 is possible for 0/1 ratios within the conversion period. For 256/256 (100%) output, port should be used.

PWDR is initialized to H'00 by a reset, and in the standby modes, watch mode, subactiv subsleep mode, and module stop mode.

9.2.3 PWM Data Polarity Registers A and B (PWDPRA and PWDPRB)

PWDPRA

Bit	7	6	5	4	3	2	1
	OS7	OS6	OS5	OS4	OS3	OS2	OS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWDPRB

Bit	7	6	5	4	3	2	1
	OS15	OS14	OS13	OS12	OS11	OS10	OS9
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Each PWDPR is an 8-bit readable/writable register that controls the polarity of the PW Bits OS0 to OS15 correspond to outputs PW0 to PW15.

PWOERA

Bit	7	6	5	4	3	2	1
	OE7	OE6	OE5	OE4	OE3	OE2	OE1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWOERB

Bit	7	6	5	4	3	2	1
	OE15	OE14	OE13	OE12	OE11	OE10	OE9
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Each PWOER is an 8-bit readable/writable register that switches between PWM outpu
output. Bits OE15 to OE0 correspond to outputs PW15 to PW0. To set a pin in the ou
setting in the port direction register is also necessary. Bits P17DDR to P10DDR corre
outputs PW7 to PW0, and bits P27DDR to P20DDR correspond to outputs PW15 to P

PWOER is initialized to H'00 by a reset and in hardware standby mode.

DDR	OE	Description
0	0	Port input (Ini
	1	Port input
1	0	Port output or PWM 256/256 output
	1	PWM output (0 to 255/256 output)

PCSR is an 8-bit readable/writable register that selects the PWM timer input clock.

PCSR is initialized to H'00 by a reset, and in hardware standby mode.

Bits 7 to 3—Reserved: These bits cannot be modified and are always read as 0.

Bits 2 and 1—PWM Clock Select (PWCKB, PWCKA): Together with bits PWCKE
PWCKS in PWSL, these bits select the internal clock input to TCNT in the PWM timer
details, see section 9.2.1, PWM Register Select (PWSL).

Bit 0—Reserved: Do not set this bit to 1.

9.2.6 Port 1 Data Direction Register (P1DDR)

Bit	7	6	5	4	3	2	1
	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

P1DDR is an 8-bit write-only register that specifies the input/output direction and PWM
each pin of port 1 on a bit-by-bit basis.

Port 1 pins are multiplexed with pins PW0 to PW7. The bit corresponding to a pin to be
PWM output should be set to 1.

For details on P1DDR, see section 8.2, Port 1.

each pin of port J on a bit-by-bit basis.

Port 2 pins are multiplexed with pins PW8 to PW15. The bit corresponding to a pin to
PWM output should be set to 1.

For details on P2DDR, see section 8.3, Port 2.

9.2.8 Port 1 Data Register (P1DR)

Bit	7	6	5	4	3	2	1
	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P1DR is an 8-bit readable/writable register used to fix PWM output at 1 (when OS = 0
OS = 1).

For details on P1DR, see section 8.2, Port 1.

9.2.9 Port 2 Data Register (P2DR)

Bit	7	6	5	4	3	2	1
	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

P2DR is an 8-bit readable/writable register used to fix PWM output at 1 (when OS = 0
OS = 1).

For details on P2DR, see section 8.3, Port 2.

MSTPCR comprises two 8-bit readable/writable registers, and is used to perform modu
mode control.

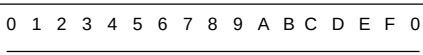
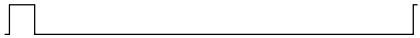
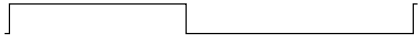
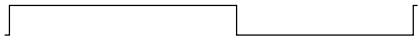
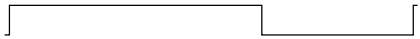
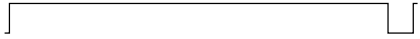
When the MSTP11 bit is set to 1, 8-bit PWM timer operation is halted and a transition
module stop mode. For details, see section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini
software standby mode.

MSTPCRH Bit 3—Module Stop (MSTP11): Specifies PWM module stop mode.

MSTPCRH Bit 3

MSTP11	Description
0	PWM module stop mode is cleared
1	PWM module stop mode is set (In

Upper 4 Bits	Basic Pulse Waveform (Internal)
0000	
0001	
0010	
0011	
0100	
0101	
0110	
0111	
⋮	
1000	
1001	
1010	
1011	
1100	
1101	
1110	
1111	

0000													
0001													
0010								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

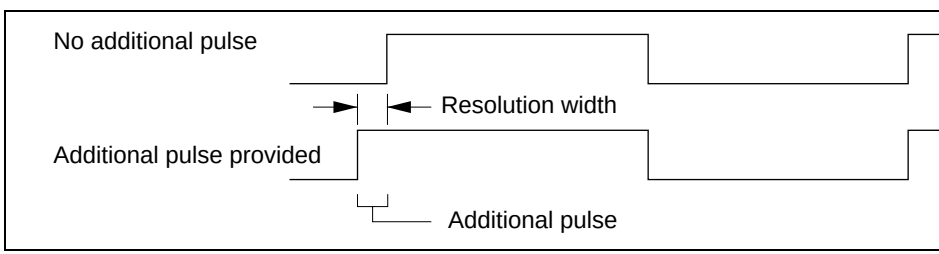


Figure 9.2 Example of Additional Pulse Timing (when Upper 4 Bits of PWDR

Both channels share the same counter (DACNT) and control register (DACR).

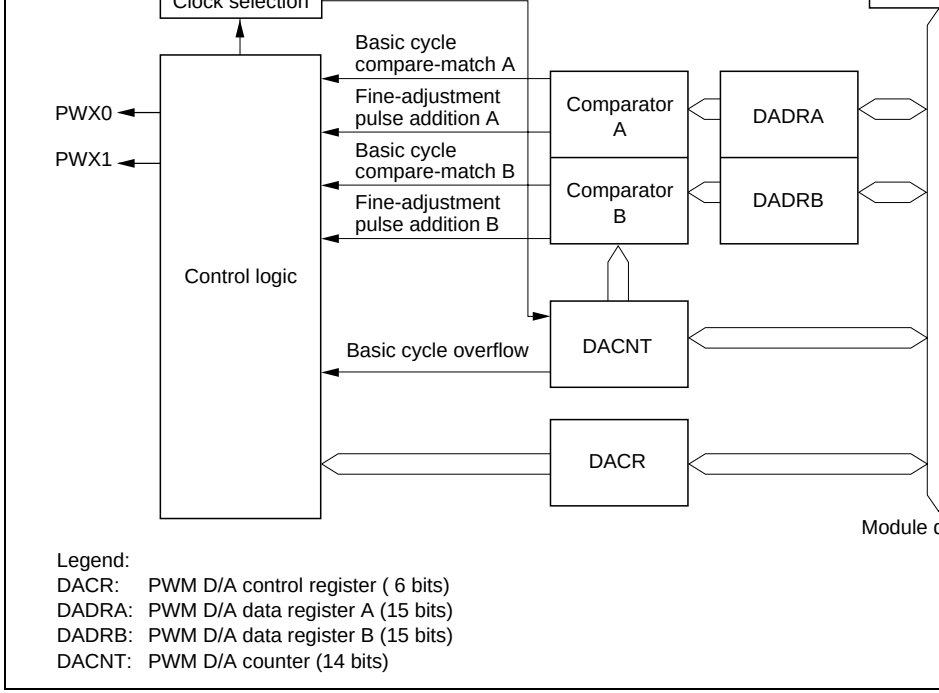
10.1.1 Features

The features of the 14-bit PWM (D/A) are listed below.

- The pulse is subdivided into multiple base cycles to reduce ripple.
- Two resolution settings and two base cycle settings are available

  The resolution can be set equal to one or two system clock cycles. The base cycle
  equal to T × 64 or T × 256, where T is the resolution.
- Four operating rates

  The two resolution settings and two base cycle settings combine to give a selectio
  operating rates.



Legend:
DACR: PWM D/A control register (6 bits)
DADRA: PWM D/A data register A (15 bits)
DADRB: PWM D/A data register B (15 bits)
DACNT: PWM D/A counter (14 bits)

Figure 10.1 PWM D/A Block Diagram

10.1.4 Register Configuration

Table 10.2 lists the registers of the PWM (D/A) module.

Table 10.2 Register Configuration

Name	Abbreviation	R/W	Initial value	A
PWM D/A control register	DACR	R/W	H'30	H
PWM D/A data register A high	DADRAH	R/W	H'FF	H
PWM D/A data register A low	DADRAL	R/W	H'FF	H
PWM D/A data register B high	DADRBH	R/W	H'FF	H
PWM D/A data register B low	DADRBL	R/W	H'FF	H
PWM D/A counter high	DACNTH	R/W	H'00	H
PWM D/A counter low	DACNTL	R/W	H'03	H
Module stop control register	MSTPCRH	R/W	H'3F	H
	MSTPCRL	R/W	H'FF	H

Notes: 1. Lower 16 bits of the address.

2. Registers in the 14-bit PWM timer are assigned to the same addresses as t registers. In this case, register selection is performed by the IICE bit in the s control register (STCR), and also the same addresses are shared by DADF DACR, and by DADRB and DACNT. Switching is performed by the REGS b or DADRB.

Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

DACNT is a 14-bit readable/writable up-counter that increments on an input clock puls
input clock is selected by the clock select bit (CKS) in DACR. The CPU can read and v
DACNT value, but since DACNT is a 16-bit register, data transfers between it and the
performed using a temporary register (TEMP). See section 10.3, Bus Master Interface,

DACNT functions as the time base for both PWM (D/A) channels. When a channel ope
14-bit precision, it uses all DACNT bits. When a channel operates with 12-bit precision
lower 12 (counter) bits and ignores the upper two (counter) bits.

DACNT is initialized to H'0003 by a reset, in the standby modes, watch mode, subactiv
subsleep mode, and module stop mode, and by the PWME bit.

Bit 1 of DACNTL (CPU) is not used, and is always read as 1.

DACNTL Bit 0—Register Select (REGS): DADRA and DACR, and DADRB and DA
located at the same addresses. The REGS bit specifies which registers can be accessed.
bit can be accessed regardless of whether DADRB or DACNT is selected.

Bit 0

REGS	Description
0	DADRA and DADRB can be accessed
1	DACR and DACNT can be accessed (In

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
DADRB	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

There are two 16-bit readable/writable D/A data registers: DADRA and DADRB. DA corresponds to PWM (D/A) channel A, and DADRB to PWM (D/A) channel B. The C and write the PWM (D/A) data register values, but since DADRA and DADRB are 16 registers, data transfers between them and the CPU are performed using a temporary r (TEMP). See section 10.3, Bus Master Interface, for details.

The least significant (CPU) bit of DADRA is not used and is always read as 1.

DADR is initialized to H'FFFF by a reset, and in the standby modes, watch mode, sub subsleep mode, and module stop mode.

Bits 15 to 3—PWM D/A Data 13 to 0 (DA13 to DA0): The digital value to be conve analog value is set in the upper 14 bits of the PWM (D/A) data register.

In each base cycle, the DACNT value is continually compared with these upper 14 bit determine the duty cycle of the output waveform, and to decide whether to output a fi adjustment pulse equal in width to the resolution. To enable this operation, the data re be set within a range that depends on the carrier frequency select bit (CFS). If the DA outside this range, the PWM output is held constant.

A channel can be operated with 12-bit precision by keeping the two lowest data bits (I DA1) cleared to 0 and writing the data to be converted in the upper 12 bits. The two lo bits correspond to the two highest counter (DACNT) bits.

DADRA Bit 0—Reserved: This bit cannot be modified and is always read as 1.

DADRB Bit 0—Register Select (REGS): DADRA and DACR, and DADRB and DA located at the same addresses. The REGS bit specifies which registers can be accessed. bit can be accessed regardless of whether DADRB or DACNT is selected.

Bit 0

REGS	Description
0	DADRA and DADRB can be accessed
1	DACR and DACNT can be accessed (In

10.2.3 PWM D/A Control Register (DACR)

Bit	7	6	5	4	3	2	1	
	TEST	PWME	—	—	OEB	OEA	OS	
Initial value	0	0	1	1	0	0	0	
Read/Write	R/W	R/W	—	—	R/W	R/W	R/W	

DACR is an 8-bit readable/writable register that selects test mode, enables the PWM ou selects the output phase and operating speed.

DACR is initialized to H'30 by a reset, and in the standby modes, watch mode, subactiv subsleep mode, and module stop mode.

Bit 7—Test Mode (TEST): Selects test mode, which is used in testing the chip. Norma should be cleared to 0.

PWME	Description
0	DACNT operates as a 14-bit up-counter
1	DACNT halts at H'0003

Bits 5 and 4—Reserved: These bits cannot be modified and are always read as 1.

Bit 3—Output Enable B (OEB): Enables or disables output on PWM (D/A) channel

Bit 3

OEB	Description
0	PWM (D/A) channel B output (at the PWX1 pin) is disabled
1	PWM (D/A) channel B output (at the PWX1 pin) is enabled

Bit 2—Output Enable A (OEA): Enables or disables output on PWM (D/A) channel

Bit 2

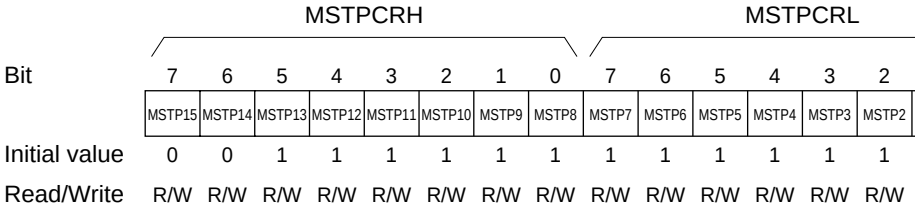
OEA	Description
0	PWM (D/A) channel A output (at the PWX0 pin) is disabled
1	PWM (D/A) channel A output (at the PWX0 pin) is enabled

Bit 1—Output Select (OS): Selects the phase of the PWM (D/A) output.

Bit 1

OS	Description
0	Direct PWM output
1	Inverted PWM output

10.2.4 Module Stop Control Register (MSTPCR)



MSTPCR comprises two 8-bit readable/writable registers, and is used to perform modu mode control.

When the MSTP11 bit is set to 1, 14-bit PWM timer operation is halted and a transition module stop mode. For details, see section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

MSTPCRH Bit 3—Module Stop (MSTP11): Specifies PWMX module stop mode.

MSTPCRH Bit 3	
MSTP11	**Description**
0	PWMX module stop mode is cleared
1	PWMX module stop mode is set (In

When the upper byte is written, the upper-byte write data is stored in TEMP. Next lower byte is written, the lower-byte write data and TEMP value are combined, an combined 16-bit value is written in the register.

- Read

  When the upper byte is read, the upper-byte value is transferred to the CPU and th value is transferred to TEMP. Next, when the lower byte is read, the lower-byte va TEMP is transferred to the CPU.

These registers should always be accessed 16 bits at a time using an MOV instruction access or two consecutive byte accesses), and the upper byte should always be accesse lower byte. Correct data will not be transferred if only the upper byte or only the lowe accessed. Also note that a bit-manipulation instruction cannot be used to access these

Figure 10.2 shows the data flow for access to DACNT. The other registers are accesse

Example 1: Write to DACNT

```
MOV.W R0, @DACNT     ; Write R0 contents to DACNT
```

Example 2: Read DADRA

```
MOV.W @DADRA, R0     ; Transfer contents of DADRA to R0
```

(first) and lower byte (second).

×: This type of access may give incorrect results.

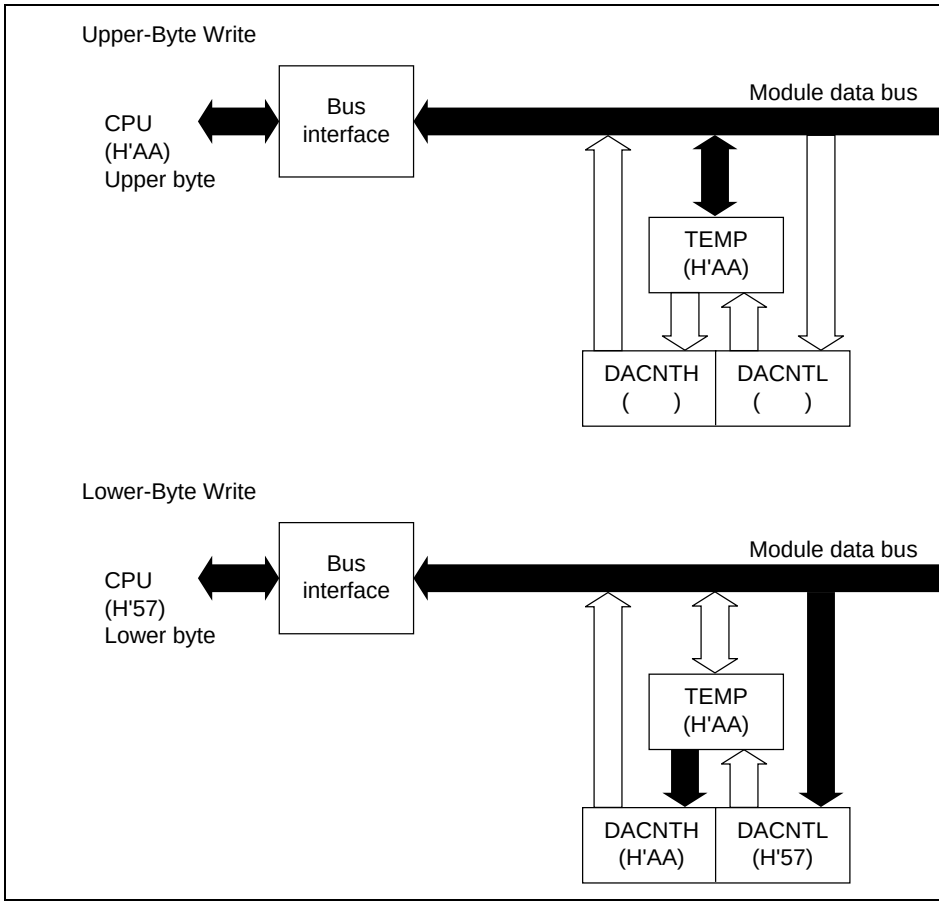


Figure 10.2 (a) Access to DACNT (CPU Writes H'AA57 to DACNT)

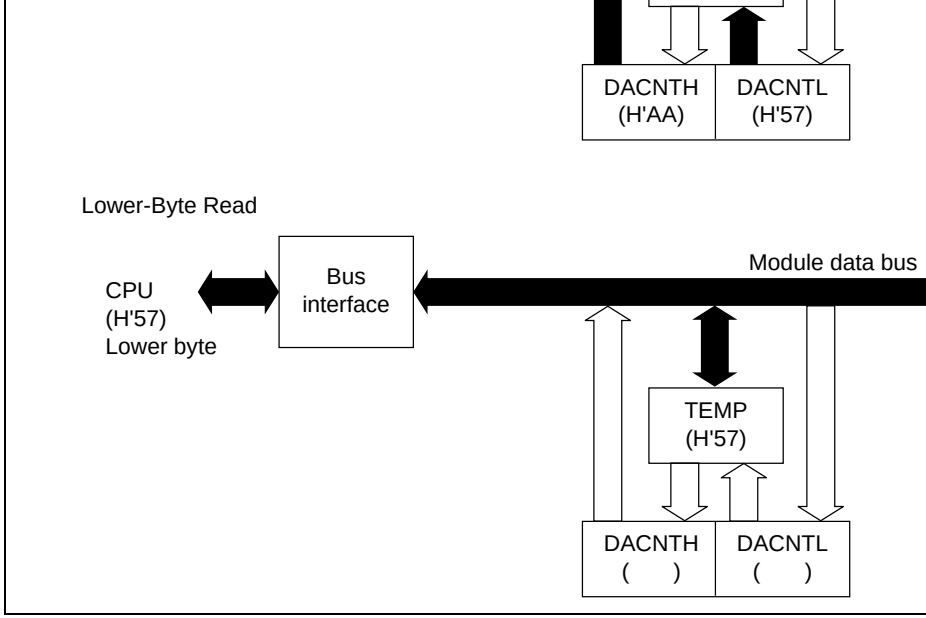
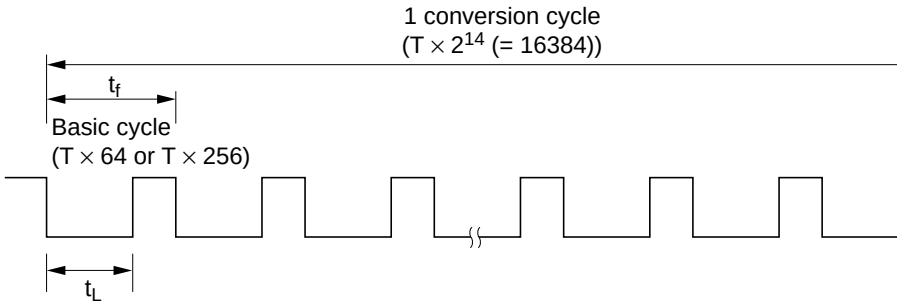


Figure 10.2 (b) Access to DACNT (CPU Reads H'AA57 from DACNT



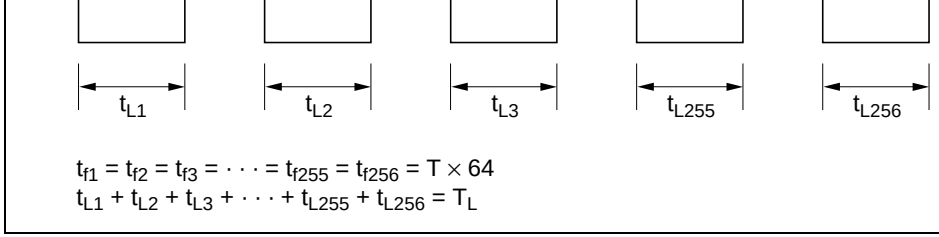
Legend:
T: Resolution
$T_L = \sum_{n=1}^{m} t_{Ln}$ (when OS = 0)
(When CFS = 0, m = 256; when CFS = 1, m = 64)

Figure 10.3 PWM D/A Operation

Table 10.4 summarizes the relationships of the CKS, CFS, and OS bit settings to the re base cycle, and conversion cycle. The PWM output remains flat unless DADR contains certain minimum value. Table 10.4 indicates the range of DADR settings that give an o waveform like the one in figure 10.3, and lists the conversion cycle length when low-or bits are kept cleared to 0, reducing the conversion precision to 12 bits or 10 bits.

					2. (Data value) × T (DADR = H'0401 to H'FFFD)	12			0	0
						10	0	0	0	0
		1	25.6	1638.4	1. Always low (or high) level output (DADR = H'0003 to H'00FF)	14				
					2. (Data value) × T (DADR = H'0103 to H'FFFF)	12			0	0
						10	0	0	0	0
1	0.2	0	12.8	3276.8	1. Always low (or high) level output (DADR = H'0001 to H'03FD)	14				
					2. (Data value) × T (DADR = H'0401 to H'FFFD)	12			0	0
						10	0	0	0	0
		1	51.2	3276.8	1. Always low (or high) level output (DADR = H'0003 to H'00FF)	14				
					2. (Data value) × T (DADR = H'0103 to H'FFFF)	12			0	0
						10	0	0	0	0

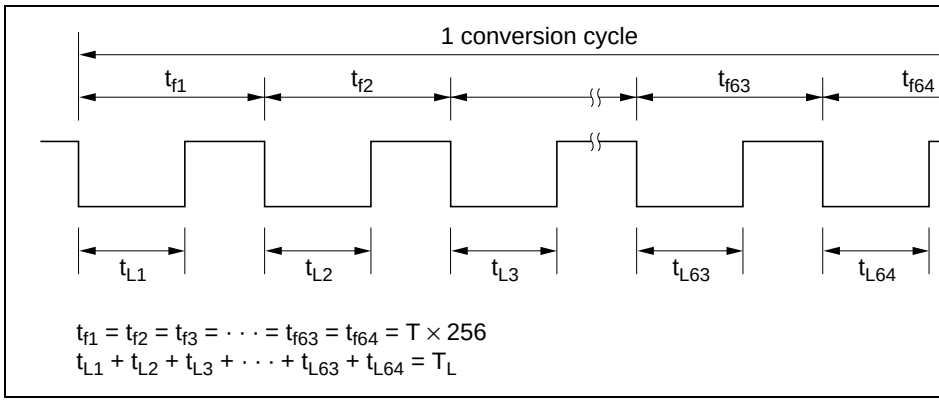
Note: * This column indicates the conversion cycle when specific DADR bits are fix



$t_{f1} = t_{f2} = t_{f3} = \cdots = t_{f255} = t_{f256} = T \times 64$
$t_{L1} + t_{L2} + t_{L3} + \cdots + t_{L255} + t_{L256} = T_L$

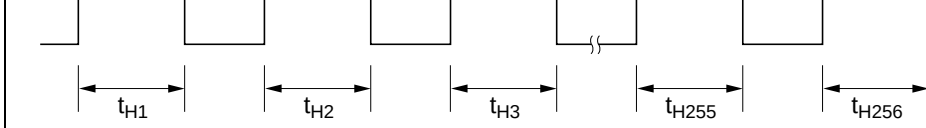
Figure 10.4 (1) Output Waveform

b. CFS = 1 [base cycle = resolution (T) × 256]



$t_{f1} = t_{f2} = t_{f3} = \cdots = t_{f63} = t_{f64} = T \times 256$
$t_{L1} + t_{L2} + t_{L3} + \cdots + t_{L63} + t_{L64} = T_L$

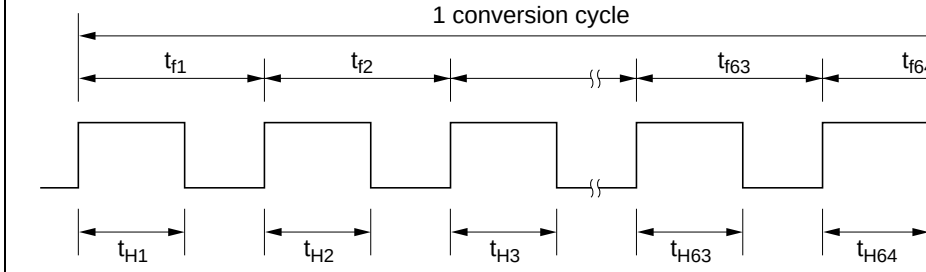
Figure 10.4 (2) Output Waveform



$t_{f1} = t_{f2} = t_{f3} = \cdots = t_{f255} = t_{f256} = T \times 64$
$t_{H1} + t_{H2} + t_{H3} + \cdots + t_{H255} + t_{H256} = T_H$

Figure 10.4 (3) Output Waveform

b. CFS = 1 [base cycle = resolution (T) × 256]



$t_{f1} = t_{f2} = t_{f3} = \cdots = t_{f63} = t_{f64} = T \times 256$
$t_{H1} + t_{H2} + t_{H3} + \cdots + t_{H63} + t_{H64} = T_H$

Figure 10.4 (4) Output Waveform

external clock periods.

11.1.1 Features

The features of the free-running timer module are listed below.

- Selection of four clock sources
  - The free-running counter can be driven by an internal clock source (φ/2, φ/8, o external clock input (enabling use as an external event counter).
- Two independent comparators
  - Each comparator can generate an independent waveform.
- Four input capture channels
  - The current count can be captured on the rising or falling edge (selectable) of a signal.
  - The four input capture registers can be used separately, or in a buffer mode.
- Counter can be cleared under program control
  - 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 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
  - The contents of ICRD can be added automatically to the contents of OCRDM x input capture operations in this interval to be restricted.

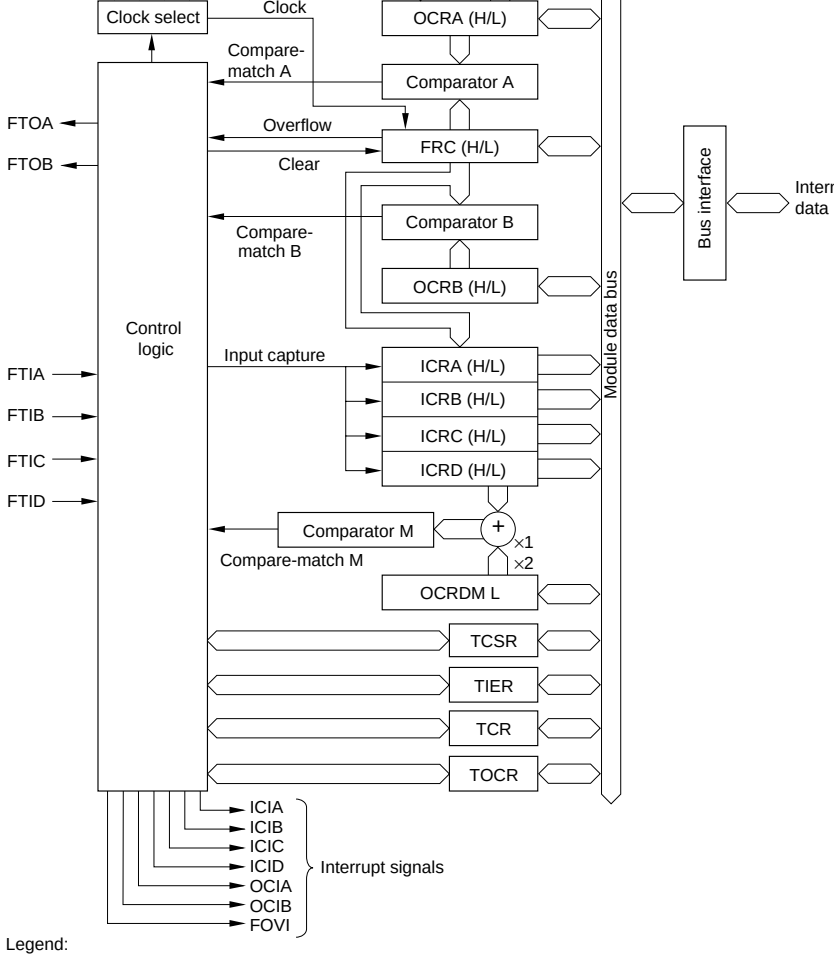


Figure 11.1 Block Diagram of 16-Bit Free-Running Timer

Output compare B	FTOB	Output	Output compare B output
Input capture A	FTIA	Input	Input capture A input
Input capture B	FTIB	Input	Input capture B input
Input capture C	FTIC	Input	Input capture C input
Input capture D	FTID	Input	Input capture D input

Free-running counter	FRC	R/W	H'0000	H'FF
Output compare register A	OCRA	R/W	H'FFFF	H'FF
Output compare register B	OCRB	R/W	H'FFFF	H'FF
Timer control register	TCR	R/W	H'00	H'FF
Timer output compare control register	TOCR	R/W	H'00	H'FF
Input capture register A	ICRA	R	H'0000	H'FF
Input capture register B	ICRB	R	H'0000	H'FF
Input capture register C	ICRC	R	H'0000	H'FF
Input capture register D	ICRD	R	H'0000	H'FF
Output compare register AR	OCRAR	R/W	H'FFFF	H'FF
Output compare register AF	OCRAF	R/W	H'FFFF	H'FF
Output compare register DM	OCRDM	R/W	H'0000	H'FF
Module stop control register	MSTPCRH	R/W	H'3F	H'FF
	MSTPCRL	R/W	H'FF	H'FF

Notes: 1. Lower 16 bits of the address.

2. Bits 7 to 1 are read-only; only 0 can be written to clear the flags.
   Bit 0 is readable/writable.
3. OCRA and OCRB share the same address. Access is controlled by the OCR bit in TOCR.
4. ICRA, ICRB, and ICRC share the same addresses with OCRAR, OCRAF, a OCRDM. Access is controlled by the ICRS bit in TOCR.

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

FRC is a 16-bit readable/writable up-counter that increments on an internal pulse gene
clock source. The clock source is selected by bits CKS1 and CKS0 in TCR.

FRC can also be cleared by compare-match A.

When FRC overflows from H'FFFF to H'0000, the overflow flag (OVF) in TCSR is se

FRC is initialized to H'0000 by a reset and in hardware standby mode.

11.2.2 Output Compare Registers A and B (OCRA, OCRB)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCRA and OCRB are 16-bit readable/writable registers, the contents of which are con
compared with the value in the FRC. When a match is detected, the corresponding out
flags (OCFA or OCFB) is set in TCSR.

In addition, if the output enable bit (OEA or OEB) in TOCR is set to 1, when OCR an
values match, the logic level selected by the output level bit (OLVLA or OLVLB) in T
output at the output compare pin (FTOA or FTOB). Following a reset, the FTOA and
output levels are 0 until the first compare-match.

OCR is initialized to H'FFFF by a reset and in hardware standby mode.

When the rising or falling edge of the signal at an input capture input pin (FTIA to FTI
detected, the current FRC value is copied to the corresponding input capture register (I
ICRD). At the same time, the corresponding input capture flag (ICFA to ICFD) in TCS
1. The input capture edge is selected by the input edge select bits (IEDGA to IEDGD) i

ICRC and ICRD can be used as ICRA and ICRB buffer registers, respectively, and mac
perform buffer operations, by means of buffer enable bits A and B (BUFEA, BUFEB) i

Figure 11.2 shows the connections when ICRC is specified as the ICRA buffer register
1). When ICRC is used as the ICRA buffer, both rising and falling edges can be specifi
transitions of the external input signal by setting IEDGA ≠ IEDGC. When IEDGA = IE
either the rising or falling edge is designated. See table 11.3.

Note: The FRC contents are transferred to the input capture register regardless of the
the input capture flag (ICF).

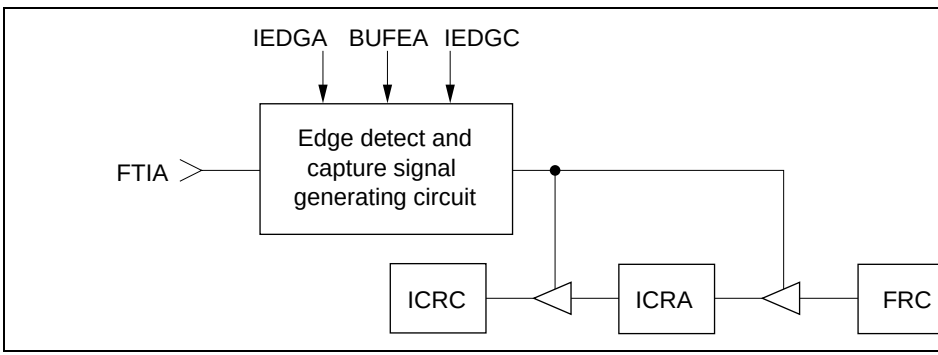


Figure 11.2 Input Capture Buffering (Example)

To ensure input capture, the width of the input capture pulse should be at least 1.5 sys periods (1.5ϕ). When triggering is enabled on both edges, the input capture pulse widt at least 2.5 system clock periods (2.5ϕ).

ICR is initialized to H'0000 by a reset and in hardware standby mode.

11.2.4 Output Compare Registers AR and AF (OCRAR, OCRAF)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCRAR and OCRAF are 16-bit readable/writable registers.

When the OCRAMS bit in TOCR is set to 1, the operation of OCRA is changed to inc of OCRAR and OCRAF. The contents of OCRAR and OCRAF are automatically add alternately to OCRA, and the result is written to OCRA. The write operation is perfor occurrence of compare-match A. In the first compare-match A after the OCRAMS bit OCRAF is added.

The operation due to compare-match A varies according to whether the compare-matc addition of OCRAR or OCRAF. The value of the OLVLA bit in TOCR is ignored, an on a compare-match A following addition of OCRAF, while 0 is output on a compare following addition of OCRAR.

When the OCRA automatically addition function is used, do not set internal clock ϕ/2 counter input clock together with an OCRAR (or OCRAF) value of H'0001 or less.

OCRAR and OCRAF are initialized to H'FFFF by a reset and in hardware standby mo

When the ICRDMS bit in TOCR is set to 1 and the contents of OCRDM are other than
the operation of ICRD is changed to include the use of OCRDM. The point at which in
D occurs is taken as the start of a mask interval. Next, twice the contents of OCRDM is
the contents of ICRD, and the result is compared with the FRC value. The point at whi
values match is taken as the end of the mask interval. New input capture D events are d
during the mask interval.

A mask interval is not generated when the ICRDMS bit is set to 1 and the contents of O
H'0000.

OCRDM is initialized to H'0000 by a reset and in hardware standby mode.

11.2.6 Timer Interrupt Enable Register (TIER)

Bit	7	6	5	4	3	2	1	
	ICIAE	ICIBE	ICICE	ICIDE	OCIAE	OCIBE	OVIE	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TIER is an 8-bit readable/writable register that enables and disables interrupts.

TIER is initialized to H'01 by a reset and in hardware standby mode.

Bit 7—Input Capture Interrupt A Enable (ICIAE): Selects whether to request input
interrupt A (ICIA) when input capture flag A (ICFA) in TCSR is set to 1.

Bit 6

ICIBE	Description
0	Input capture interrupt request B (ICIB) is disabled
1	Input capture interrupt request B (ICIB) is enabled

Bit 5—Input Capture Interrupt C Enable (ICICE): Selects whether to request inpu
interrupt C (ICIC) when input capture flag C (ICFC) in TCSR is set to 1.

Bit 5

ICICE	Description
0	Input capture interrupt request C (ICIC) is disabled
1	Input capture interrupt request C (ICIC) is enabled

Bit 4—Input Capture Interrupt D Enable (ICIDE): Selects whether to request inpu
interrupt D (ICID) when input capture flag D (ICFD) in TCSR is set to 1.

Bit 4

ICIDE	Description
0	Input capture interrupt request D (ICID) is disabled
1	Input capture interrupt request D (ICID) is enabled

Bit 3—Output Compare Interrupt A Enable (OCIAE): Selects whether to request
compare interrupt A (OCIA) when output compare flag A (OCFA) in TCSR is set to 1

Bit 3

OCIAE	Description
0	Output compare interrupt request A (OCIA) is disabled
1	Output compare interrupt request A (OCIA) is enabled

Bit 1—Timer Overflow Interrupt Enable (OVIE): Selects whether to request a free-
timer overflow interrupt (FOVI) when the timer overflow flag (OVF) in TCSR is set to

Bit 1		
OVIE	**Description**	
0	Timer overflow interrupt request (FOVI) is disabled	(In
1	Timer overflow interrupt request (FOVI) is enabled	

Bit 0—Reserved: This bit cannot be modified and is always read as 1.

11.2.7 Timer Control/Status Register (TCSR)

Bit	7	6	5	4	3	2	1
	ICFA	ICFB	ICFC	ICFD	OCFA	OCFB	OVF
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * Only 0 can be written in bits 7 to 1 to clear these flags.

TCSR is an 8-bit register used for counter clear selection and control of interrupt reque

TCSR is initialized to H'00 by a reset and in hardware standby mode.

Timing is described in section 11.3, Operation.

Bit 7—Input Capture Flag A (ICFA): This status flag indicates that the FRC value ha
transferred to ICRA by means of an input capture signal. When BUFEA = 1, ICFA indi
the old ICRA value has been moved into ICRC and the new FRC value has been transf
ICRA.

ICFA must be cleared by software. It is set by hardware, however, and cannot be set by

Bit 6—Input Capture Flag B (ICFB): This status flag indicates that the FRC value h
transferred to ICRB by means of an input capture signal. When BUFEB = 1, ICFB ind
the old ICRB value has been moved into ICRD and the new FRC value has been trans
ICRB.

ICFB must be cleared by software. It is set by hardware, however, and cannot be set b

Bit 6

ICFB	Description
0	[Clearing condition]
	Read ICFB when ICFB = 1, then write 0 in ICFB
1	[Setting condition]
	When an input capture signal causes the FRC value to be transferred to IC

Bit 5—Input Capture Flag C (ICFC): This status flag indicates that the FRC value h
transferred to ICRC by means of an input capture signal. When BUFEA = 1, on occur
signal transition in FTIC (input capture signal) specified by the IEDGC bit, ICFC is se
not transferred to ICRC. Therefore, in buffer operation, ICFC can be used as an extern
signal (by setting the ICICE bit to 1).

ICFC must be cleared by software. It is set by hardware, however, and cannot be set b

Bit 5

ICFC	Description
0	[Clearing condition]
	Read ICFC when ICFC = 1, then write 0 in ICFC
1	[Setting condition]
	When an input capture signal is received

ICFD	Description
0	[Clearing condition] (In Read ICFD when ICFD = 1, then write 0 in ICFD
1	[Setting condition] When an input capture signal is received

Bit 3—Output Compare Flag A (OCFA): This status flag indicates that the FRC valu the OCRA value. This flag must be cleared by software. It is set by hardware, however cannot be set by software.

Bit 3

OCFA	Description
0	[Clearing condition] (In Read OCFA when OCFA = 1, then write 0 in OCFA
1	[Setting condition] When FRC = OCRA

Bit 2—Output Compare Flag B (OCFB): This status flag indicates that the FRC valu the OCRB value. This flag must be cleared by software. It is set by hardware, however, be set by software.

Bit 2

OCFB	Description
0	[Clearing condition] (In Read OCFB when OCFB = 1, then write 0 in OCFB
1	[Setting condition] When FRC = OCRB

1	[Setting condition] When FRC changes from H'FFFF to H'0000

Bit 0—Counter Clear A (CCLRA): This bit selects whether the FRC is to be cleared match A (when the FRC and OCRA values match).

Bit 0

CCLRA	Description
0	FRC clearing is disabled (
1	FRC is cleared at compare-match A

11.2.8 Timer Control Register (TCR)

Bit	7	6	5	4	3	2	1
	IEDGA	IEDGB	IEDGC	IEDGD	BUFEA	BUFEB	CKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCR is an 8-bit readable/writable register that selects the rising or falling edge of the i signals, enables the input capture buffer mode, and selects the FRC clock source.

TCR is initialized to H'00 by a reset and in hardware standby mode

Bit 7—Input Edge Select A (IEDGA): Selects the rising or falling edge of the input signal (FTIA).

Bit 6

IEDGB	Description
0	Capture on the falling edge of FTIB (In
1	Capture on the rising edge of FTIB

Bit 5—Input Edge Select C (IEDGC): Selects the rising or falling edge of the input c signal (FTIC).

Bit 5

IEDGC	Description
0	Capture on the falling edge of FTIC (In
1	Capture on the rising edge of FTIC

Bit 4—Input Edge Select D (IEDGD): Selects the rising or falling edge of the input c signal (FTID).

Bit 4

IEDGD	Description
0	Capture on the falling edge of FTID (In
1	Capture on the rising edge of FTID

Bit 3—Buffer Enable A (BUFEA): Selects whether ICRC is to be used as a buffer reg ICRA.

Bit 3

BUFEA	Description
0	ICRC is not used as a buffer register for input capture A (In
1	ICRC is used as a buffer register for input capture A

Bits 1 and 0—Clock Select (CKS1, CKS0): Select external clock input or one of thr clock sources for the FRC. External clock pulses are counted on the rising edge of sig the external clock input pin (FTCI).

Bit 1	Bit 0	
CKS1	**CKS0**	**Description**
0	0	φ/2 internal clock source (
	1	φ/8 internal clock source
1	0	φ/32 internal clock source
	1	External clock source (rising edge)

11.2.9 Timer Output Compare Control Register (TOCR)

Bit	7	6	5	4	3	2	1
	ICRDMS	OCRAMS	ICRS	OCRS	OEA	OEB	OLVLA
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TOCR is an 8-bit readable/writable register that enables output from the output compa selects the output levels, switches access between output compare registers A and B, c ICRD and OCRA operating mode, and switches access to input capture registers A, B

TOCR is initialized to H'00 by a reset and in hardware standby mode.

Bit 7—Input Capture D Mode Select (ICRDMS): Specifies whether ICRD is used i operating mode or in the operating mode using OCRDM.

Bit 6		
OCRAMS	**Description**	
0	The normal operating mode is specified for OCRA	(In
1	The operating mode using OCRAR and OCRAF is specified for OCRA	

Bit 5—Input Capture Register Select (ICRS): The same addresses are shared by ICR OCRAR, by ICRB and OCRAF, and by ICRC and OCRDM. The ICRS bit determines registers are selected when the shared addresses are read or written to. The operation of ICRB, and ICRC is not affected.

Bit 5		
ICRS	**Description**	
0	The ICRA, ICRB, and ICRC registers are selected	(In
1	The OCRAR, OCRAF, and OCRDM registers are selected	

Bit 4—Output Compare Register Select (OCRS): OCRA and OCRB share the same When this address is accessed, the OCRS bit selects which register is accessed. This bit affect the operation of OCRA or OCRB.

Bit 4		
OCRS	**Description**	
0	The OCRA register is selected	(In
1	The OCRB register is selected	

Bit 2—Output Enable B (OEB): Enables or disables output of the output compare B (FTOB).

Bit 2

OEB	Description
0	Output compare B output is disabled (
1	Output compare B output is enabled

Bit 1—Output Level A (OLVLA): Selects the logic level to be output at the FTOA p response to compare-match A (signal indicating a match between the FRC and OCRA When the OCRAMS bit is 1, this bit is ignored.

Bit 1

OLVLA	Description
0	0 output at compare-match A (
1	1 output at compare-match A

Bit 0—Output Level B (OLVLB): Selects the logic level to be output at the FTOB p response to compare-match B (signal indicating a match between the FRC and OCRB

Bit 0

OLVLB	Description
0	0 output at compare-match B (
1	1 output at compare-match B

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mod

When the MSTP13 bit is set to 1, FRT operation is stopped at the end of the bus cycle, module stop mode is entered. For details, see section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

MSTPCRH Bit 5—Module Stop (MSTP13): Specifies the FRT module stop mode.

Bit 5	
MSTPCRH	**Description**
0	FRT module stop mode is cleared
1	FRT module stop mode is set (In

Any of three internal clocks (ϕ/2, ϕ/8, or ϕ/32) created by division of the system clock selected by making the appropriate setting in bits CKS1 and CKS0 in TCR. Figure 11 increment timing.

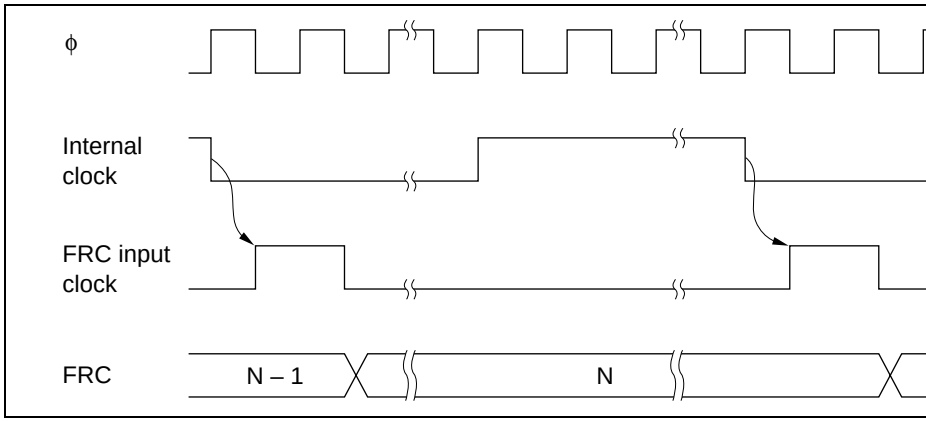


Figure 11.3 Increment Timing with Internal Clock Source

External Clock

If external clock input is selected by bits CKS1 and CKS0 in TCR, FRC increments o edge of the external clock signal.

The pulse width of the external clock signal must be at least 1.5 system clock (ϕ) perio counter will not increment correctly if the pulse width is shorter than 1.5 system clock

Figure 11.4 shows the increment timing.

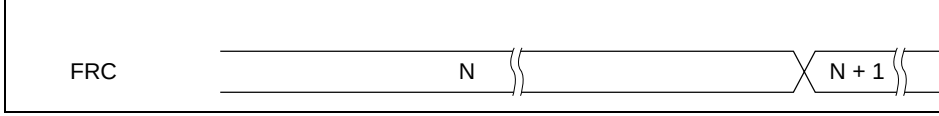


Figure 11.4 Increment Timing with External Clock Source

11.3.2 Output Compare Output Timing

When a compare-match occurs, the logic level selected by the output level bit (OLVLA OLVLB) in TOCR is output at the output compare pin (FTOA or FTOB). Figure 11.5 s timing of this operation for compare-match A.

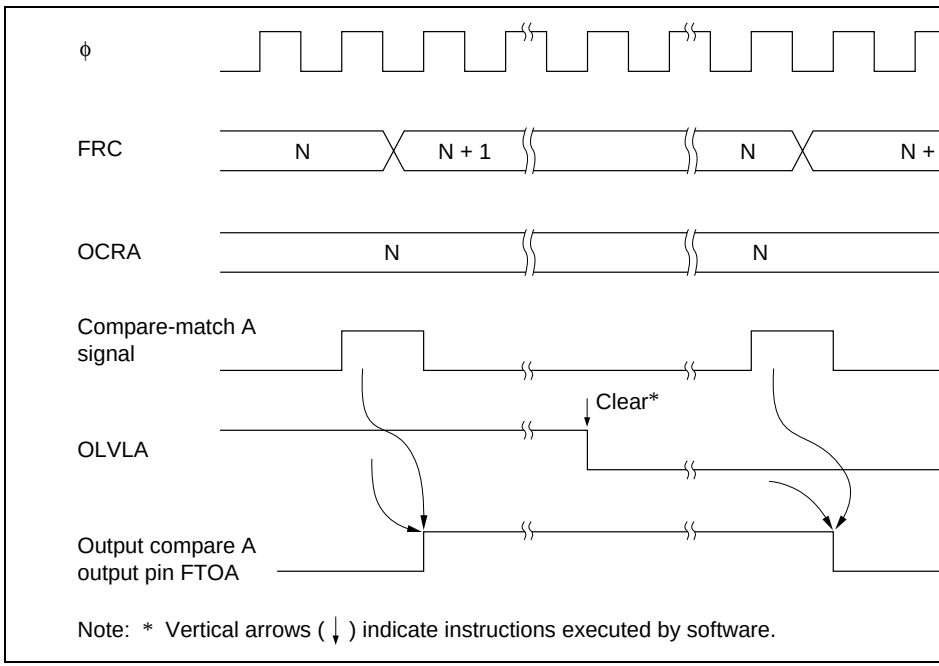


Figure 11.5 Timing of Output Compare A Output

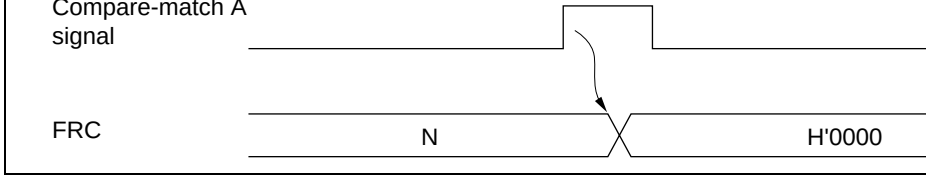


Figure 11.6 Clearing of FRC by Compare-Match A

11.3.4 Input Capture Input Timing

Input Capture Input Timing

An internal input capture signal is generated from the rising or falling edge of the sign input capture pin, as selected by the corresponding IEDGx (x = A to D) bit in TCR. Fi shows the usual input capture timing when the rising edge is selected (IEDGx = 1).

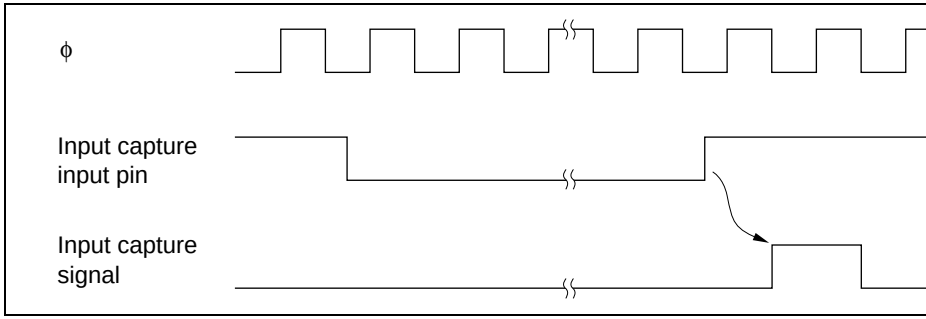


Figure 11.7 Input Capture Signal Timing (Usual Case)

If the upper byte of ICRA/B/C/D is being read when the corresponding input capture s arrives, the internal input capture signal is delayed by one system clock (ϕ) period. Fig shows the timing for this case.

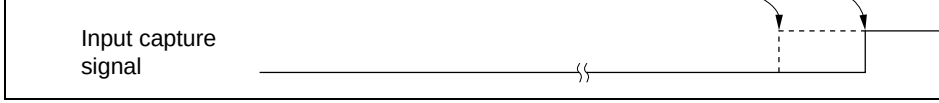


Figure 11.8 Input Capture Signal Timing (Input Capture Input when ICRA/B/C

Buffered Input Capture Input Timing

ICRC and ICRD can operate as buffers for ICRA and ICRB.

Figure 11.9 shows how input capture operates when ICRA and ICRC are used in buffe
IEDGA and IEDGC are set to different values (IEDGA = 0 and IEDGC = 1, or IEDG A
IEDGC = 0), so that input capture is performed on both the rising and falling edges of I

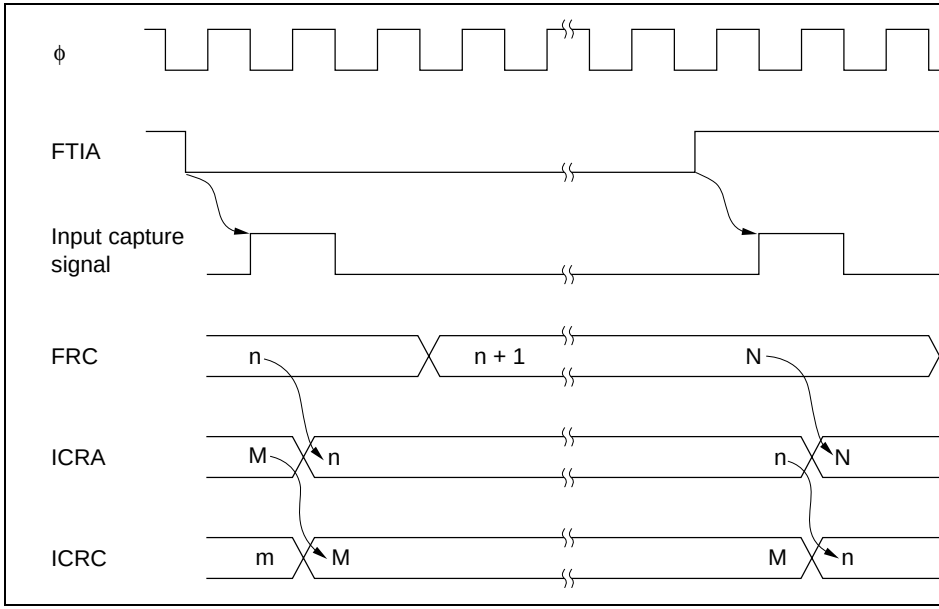


Figure 11.9 Buffered Input Capture Timing (Usual Case)

input capture is delayed by one system clock (φ) period. Figure 11.10 shows the timin
BUFEA = 1.

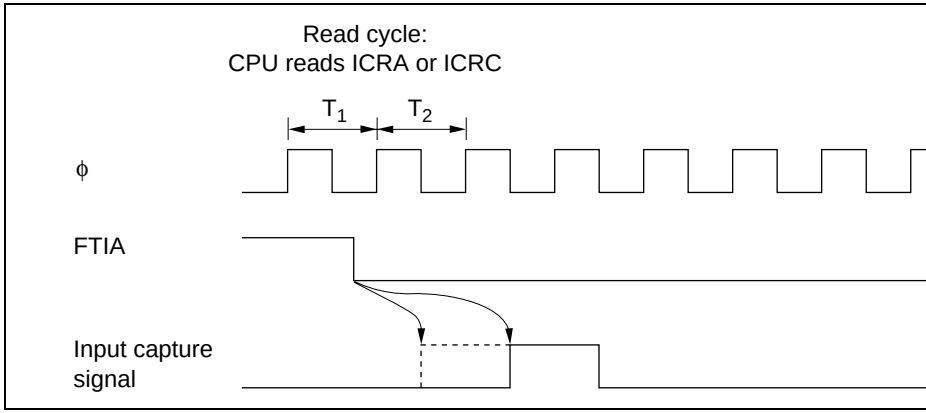


Figure 11.10 Buffered Input Capture Timing (Input Capture Input when ICRA or ICRC Is Read)

11.3.5 Timing of Input Capture Flag (ICF) Setting

The input capture flag ICFx (x = A, B, C, D) is set to 1 by the internal input capture si
FRC value is simultaneously transferred to the corresponding input capture register (I
11.11 shows the timing of this operation.

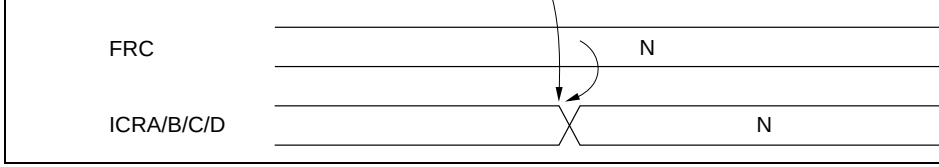


Figure 11.11 Setting of Input Capture Flag (ICFA/B/C/D)

11.3.6 Setting of Output Compare Flags A and B (OCFA, OCFB)

The output compare flags are set to 1 by an internal compare-match signal generated w
FRC value matches the OCRA or OCRB value. This compare-match signal is generate
state in which the two values match, just before FRC increments to a new value.

Accordingly, when the FRC and OCR values match, the compare-match signal is not g
until the next period of the clock source. Figure 11.12 shows the timing of the setting o
and OCFB.

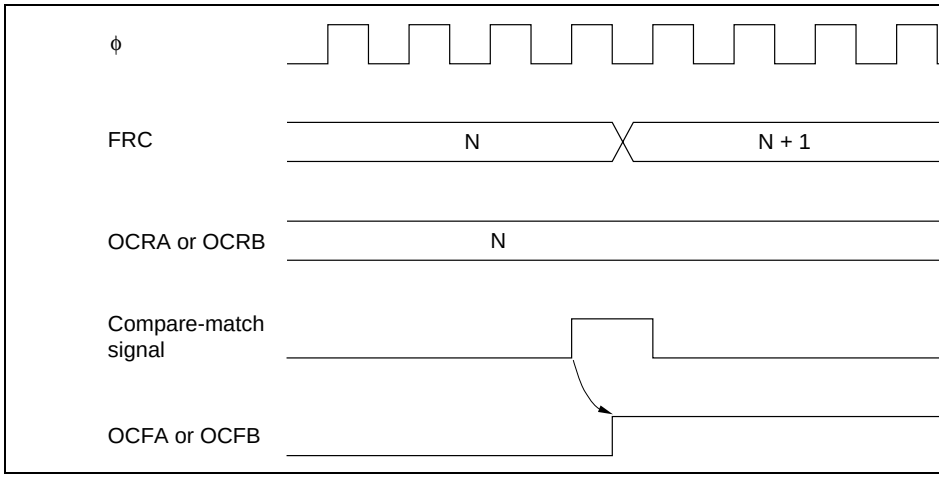


Figure 11.12 Setting of Output Compare Flag (OCFA, OCFB)

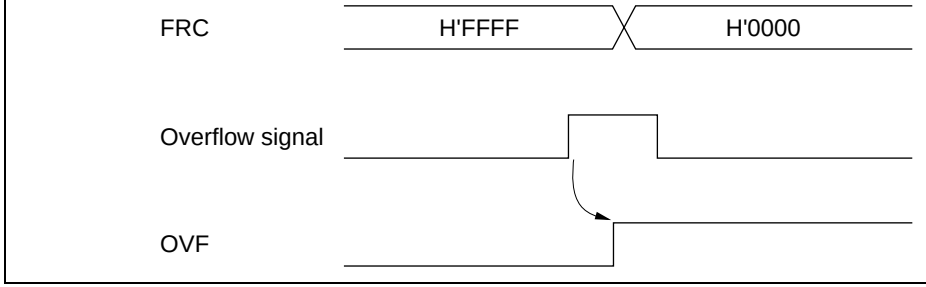


Figure 11.13 Setting of Overflow Flag (OVF)

11.3.8 Automatic Addition of OCRA and OCRAR/OCRAF

When the OCRAMS bit in TOCR is set to 1, the contents of OCRAR and OCRAF are automatically added to OCRA alternately, and when an OCRA compare-match occurs OCRA is performed. The OCRA write timing is shown in figure 11.14.

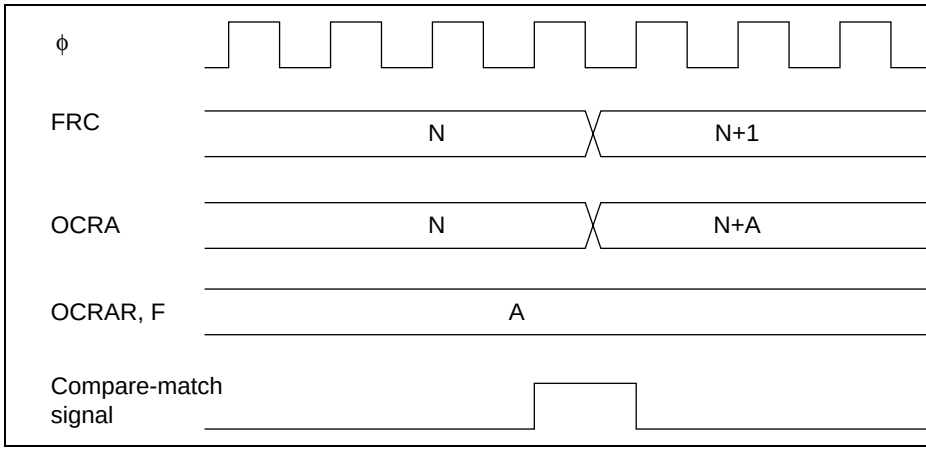


Figure 11.14 OCRA Automatic Addition Timing

an FRC compare-match. The mask signal clearing timing is shown in figure 11.16.

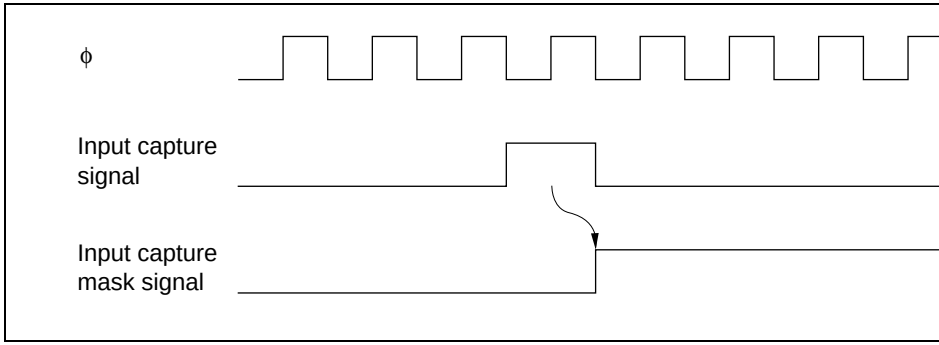


Figure 11.15 Input Capture Mask Signal Setting Timing

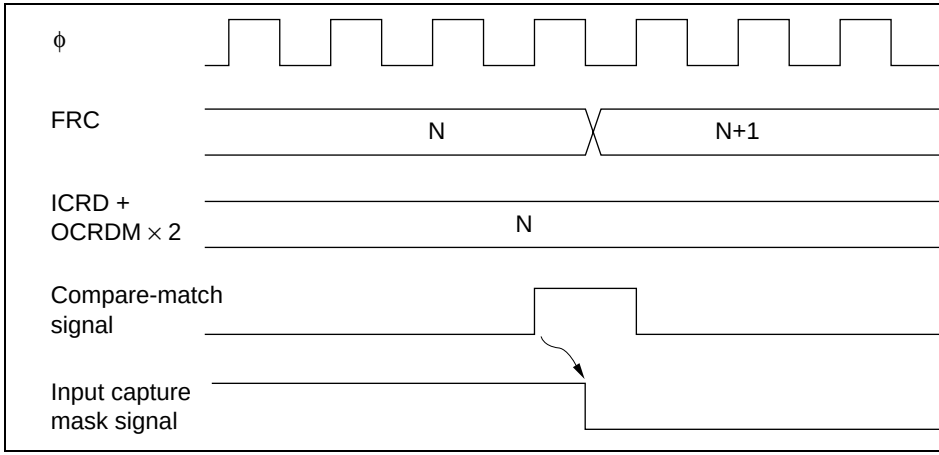


Figure 11.16 Input Capture Mask Signal Clearing Timing

Interrupt	Description	DTC Activation
ICIA	Requested by ICFA	Possible
ICIB	Requested by ICFB	Possible
ICIC	Requested by ICFC	Not possible
ICID	Requested by ICFD	Not possible
OCIA	Requested by OCFA	Possible
OCIB	Requested by OCFB	Possible
FOVI	Requested by OVF	Not possible

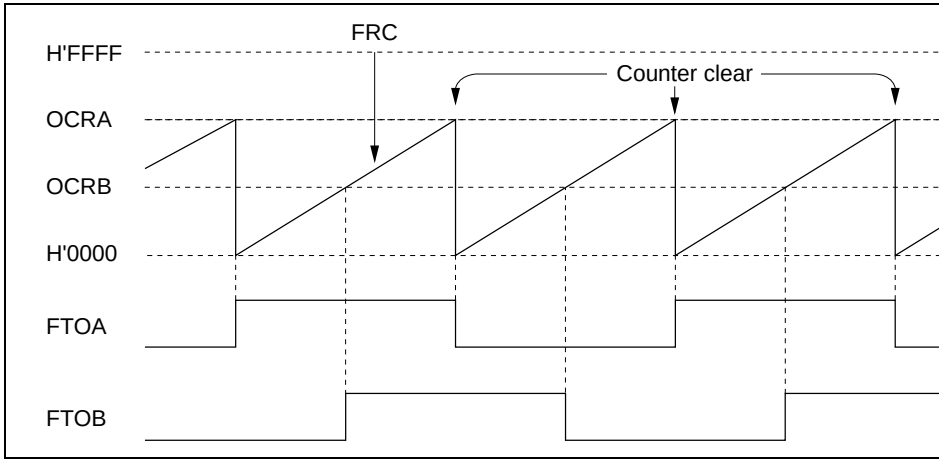


Figure 11.17 Pulse Output (Example)

signal takes priority and the write is not performed.

Figure 11.18 shows this type of contention.

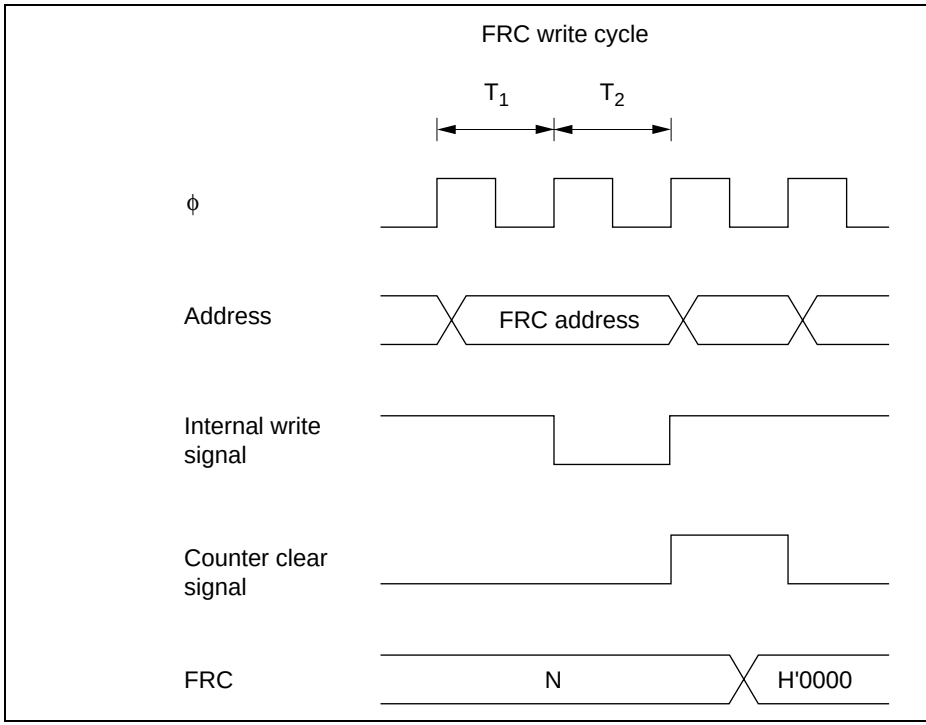


Figure 11.18 FRC Write-Clear Contention

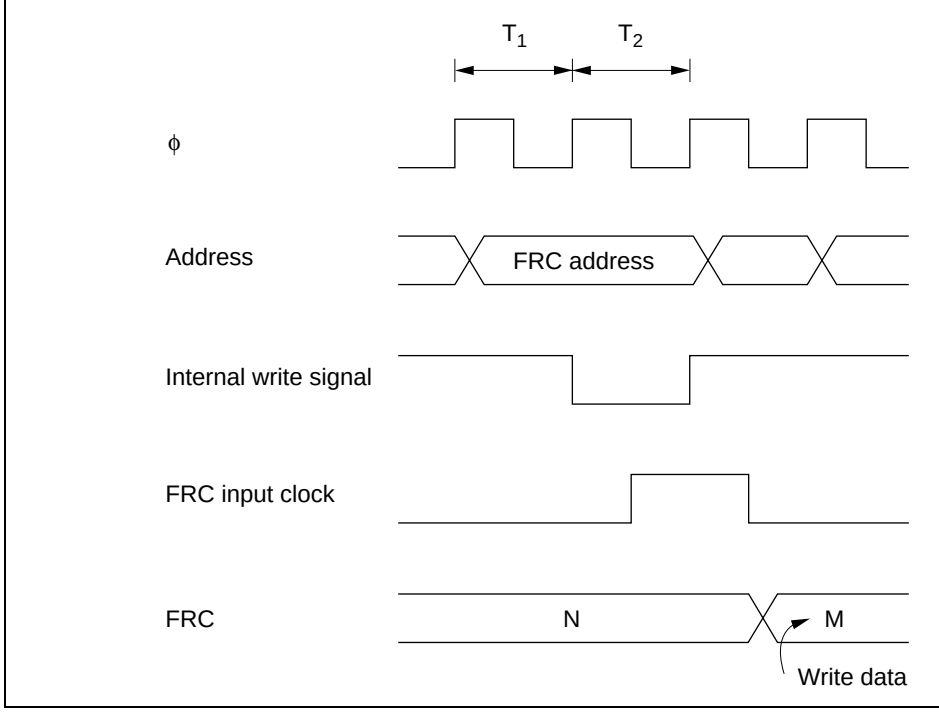


Figure 11.19 FRC Write-Increment Contention

Contention between OCR Write and Compare-Match

If a compare-match occurs during the state after an OCRA or OCRB write cycle, the w
priority and the compare-match signal is inhibited.

Figure 11.20 shows this type of contention.

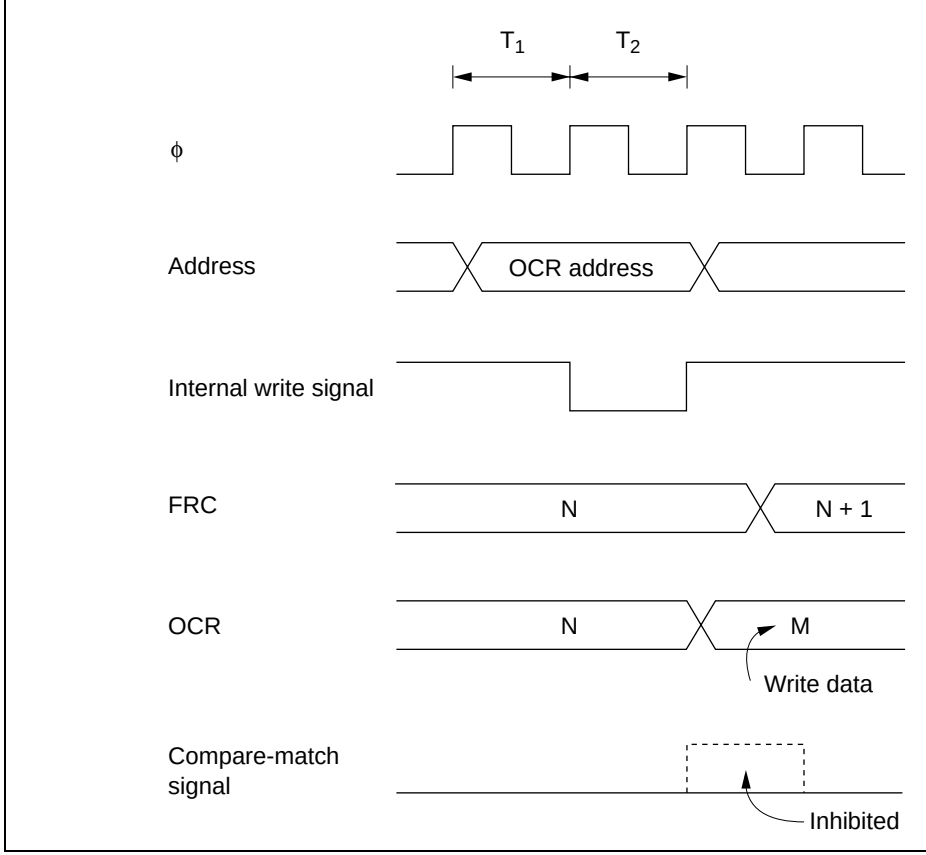


Figure 11.20 Contention between OCR Write and Compare-Match (When Automatic Addition Function Is Not Used)

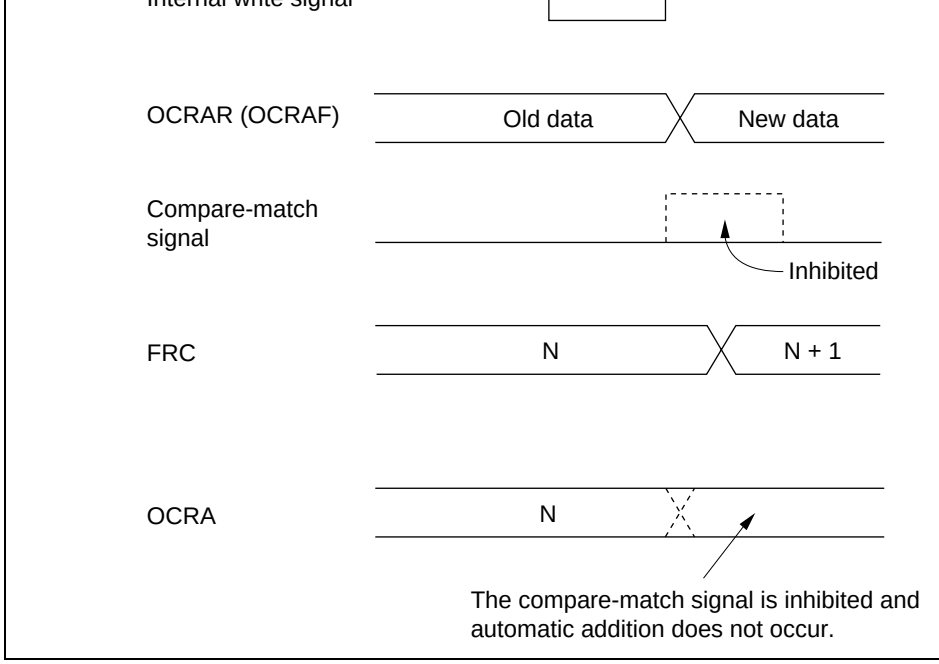
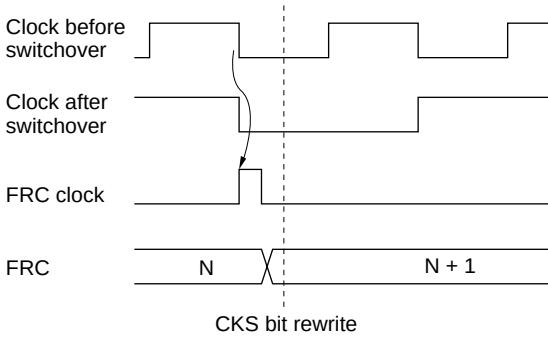
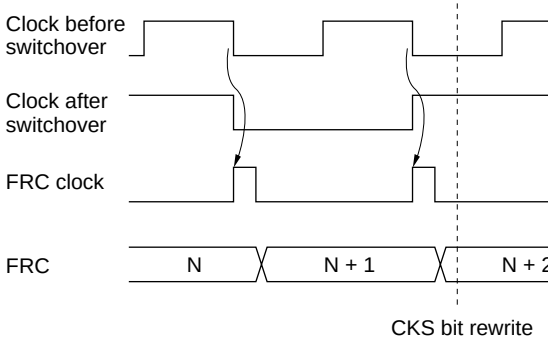
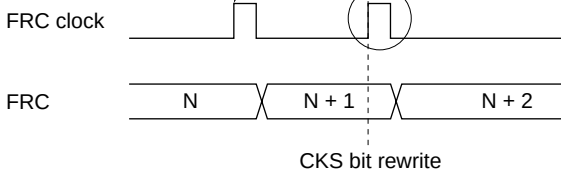


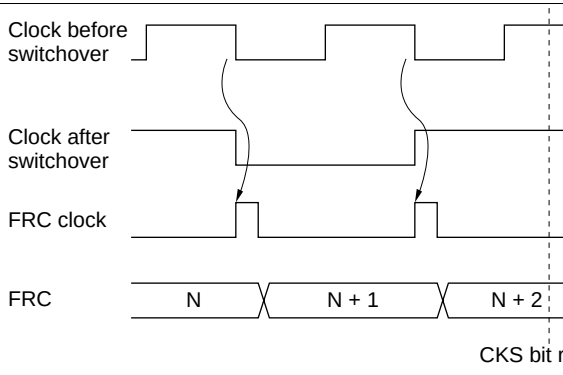
Figure 11.21 Contention between OCRAR/OCRAF Write and Compare-M (When Automatic Addition Function Is Not Used)

Switching between an internal and external clock can also cause FRC to increment.

Table 11.5 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	
2	Switching from low to high	



4	Switching from high to high	

Note: * Generated on the assumption that the switchover is a falling edge; FRC is in

be used as a multifunction timer in a variety of applications, such as generation of a re
wave output with an arbitrary duty cycle.

The H8S/2148 Group also has two similar 8-bit timer channels (TMRX and TMRY), a
H8S/2144 Group and H8S/2147N has one (TMRY). These channels can be used in a c
configuration using the timer connection function. TMRX and TMRY have greater inp
and interrupt function related restrictions than TMR0 and TMR1.

12.1.1 Features

- Selection of clock sources
  - TMR0, TMR1: The counter input clock can be selected from six internal clock
    external clock (enabling use as an external event counter).
  - TMRX, TMRY: The counter input clock can be selected from three internal clo
    external clock (enabling use as an external event counter).
- Selection of three ways to clear the counters
  - The counters can be cleared on compare-match A or B, or by an external reset
- Timer output controlled by two compare-match signals
  - The timer output signal in each channel is controlled by two independent comp
    signals, enabling the timer to be used for various applications, such as the gene
    pulse output or PWM output with an arbitrary duty cycle.
    (Note: TMRY does not have a timer output pin.)
- Cascading of the two channels (TMR0, TMR1)
  - Operation as a 16-bit timer can be performed using channel 0 as the upper half
    1 as the lower half (16-bit count mode).
  - Channel 1 can be used to count channel 0 compare-match occurrences (compar
    count mode).
- Multiple interrupt sources for each channel
  - TMR0, TMR1, TMRY: Two compare-match interrupts and one overflow inter
    requested independently.
  - TMRX: One input capture source is available.

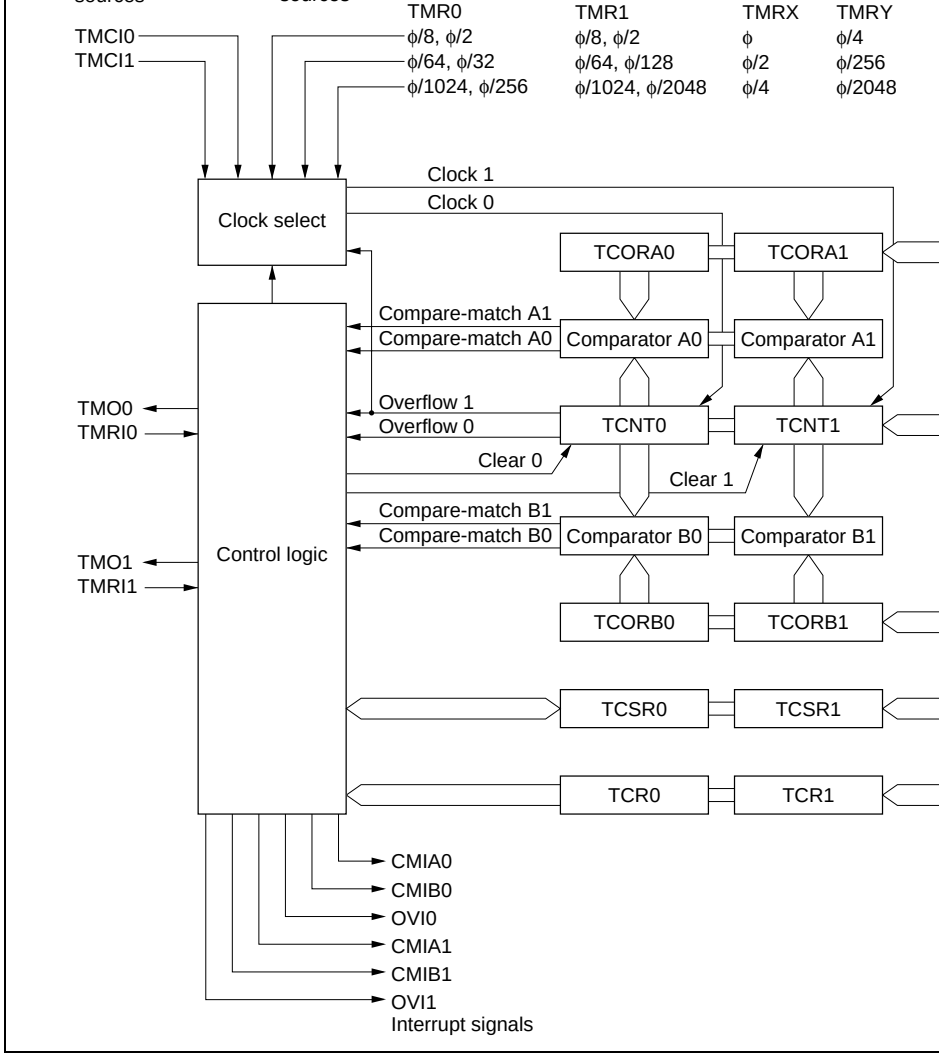


Figure 12.1 Block Diagram of 8-Bit Timer Module

	Timer reset input	TMRI0	Input	External reset input for th
1	Timer output	TMO1	Output	Output controlled by com
	Timer clock input	TMCI1	Input	External clock input for th
	Timer reset input	TMRI1	Input	External reset input for th
X	Timer output	TMOX	Output	Output controlled by com
	Timer clock/ reset input	HFBACKI/TMIX (TMCIX/TMRIX)	Input	External clock/reset inpu counter
Y	Timer clock/reset input	VSYNCI/TMIY (TMCIY/TMRIY)	Input	External clock/reset inpu counter

Note: * The abbreviations TMO, TMCI, and TMRI are used in the text, omitting the number.

Channel X and Y I/O pins have the same internal configuration as channels and therefore the same abbreviations are used.

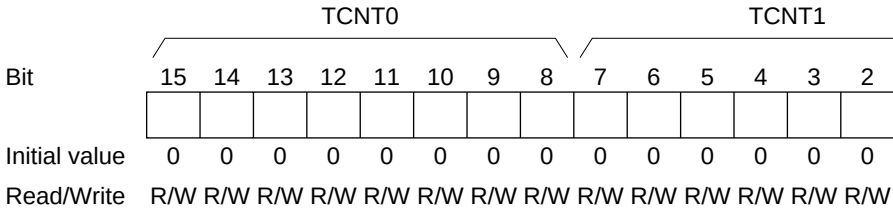
	Time constant register A0	TCORA0	R/W	H'FF	
	Time constant register B0	TCORB0	R/W	H'FF	
	Time counter 0	TCNT0	R/W	H'00	
1	Timer control register 1	TCR1	R/W	H'00	
	Timer control/status register 1	TCSR1	R/(W)*2	H'10	
	Time constant register A1	TCORA1	R/W	H'FF	
	Time constant register B1	TCORB1	R/W	H'FF	
	Timer counter 1	TCNT1	R/W	H'00	
Common	Serial/timer control register	STCR	R/W	H'00	
	Module stop control register	MSTPCRH	R/W	H'3F	
		MSTPCRL	R/W	H'FF	
	Timer connection register S	TCONRS	R/W	H'00	
X	Timer control register X	TCRX	R/W	H'00	
	Timer control/status register X	TCSRX	R/(W)*2	H'00	
	Time constant register AX	TCORAX	R/W	H'FF	
	Time constant register BX	TCORBX	R/W	H'FF	
	Timer counter X	TCNTX	R/W	H'00	
	Time constant register C	TCORC	R/W	H'FF	
	Input capture register R	TICRR	R	H'00	
	Input capture register F	TICRF	R	H'00	
Y	Timer control register Y	TCRY	R/W	H'00	
	Timer control/status register Y	TCSRY	R/(W)*2	H'00	
	Time constant register AY	TCORAY	R/W	H'FF	
	Time constant register BY	TCORBY	R/W	H'FF	
	Timer counter Y	TCNTY	R/W	H'00	
	Timer input select register	TISR	R/W	H'FE	

Notes: 1. Lower 16 bits of the address.

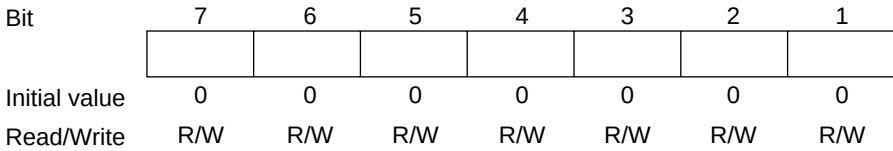
2. Only 0 can be written in bits 7 to 5, to clear these flags.

3. The abbreviations TCR, TCSR, TCORA, TCORB, and TCNT are used in the omitting the channel designation (0, 1, X, or Y).

12.2.1 Timer Counter (TCNT)



TCNTX,TCNTY



Each TCNT is an 8-bit readable/writable up-counter.

TCNT0 and TCNT1 comprise a single 16-bit register, so they can be accessed togethe access.

TCNT increments on pulses generated from an internal or external clock source. This is selected by clock select bits CKS2 to CKS0 in TCR.

TCNT can be cleared by an external reset input signal or compare-match signal. Coun CCLR1 and CCLR0 in TCR select the method of clearing.

When TCNT overflows from H'FF to H'00, the overflow flag (OVF) in TCSR is set to

The timer counters are initialized to H'00 by a reset and in hardware standby mode.

TCORAX, TCORAY

Bit	7	6	5	4	3	2	1	
Initial value	1	1	1	1	1	1	1	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TCORA is an 8-bit readable/writable register.

TCORA0 and TCORA1 comprise a single 16-bit register, so they can be accessed toge
word access.

TCORA is continually compared with the value in TCNT. When a match is detected, t
corresponding compare-match flag A (CMFA) in TCSR is set. Note, however, that com
disabled during the T2 state of a TCORA write cycle.

The timer output can be freely controlled by these compare-match signals and the settin
output select bits OS1 and OS0 in TCSR.

TCORA is initialized to H'FF by a reset and in hardware standby mode.

TCORBX, TCORBY

Bit	7	6	5	4	3	2	1
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCORB is an 8-bit readable/writable register. TCORB0 and TCORB1 comprise a sing
register, so they can be accessed together by word access.

TCORB is continually compared with the value in TCNT. When a match is detected,
corresponding compare-match flag B (CMFB) in TCSR is set. Note, however, that co
disabled during the T2 state of a TCORB write cycle.

The timer output can be freely controlled by these compare-match signals and the sett
output select bits OS3 and OS2 in TCSR.

TCORB is initialized to H'FF by a reset and in hardware standby mode.

TCR is an 8-bit readable/writable register that selects the clock source and the time at w
TCNT is cleared, and enables interrupts.

TCR is initialized to H'00 by a reset and in hardware standby mode.

For details of the timing, see section 12.3, Operation.

Bit 7—Compare-Match Interrupt Enable B (CMIEB): Selects whether the CMFB i
request (CMIB) is enabled or disabled when the CMFB flag in TCSR is set to 1.

Note that a CMIB interrupt is not requested by TMRX, regardless of the CMIEB value.

Bit 7

CMIEB	Description	
0	CMFB interrupt request (CMIB) is disabled	(In
1	CMFB interrupt request (CMIB) is enabled	

Bit 6—Compare-Match Interrupt Enable A (CMIEA): Selects whether the CMFA i
request (CMIA) is enabled or disabled when the CMFA flag in TCSR is set to 1.

Note that a CMIA interrupt is not requested by TMRX, regardless of the CMIEA value

Bit 6

CMIEA	Description	
0	CMFA interrupt request (CMIA) is disabled	(In
1	CMFA interrupt request (CMIA) is enabled	

1	OVF interrupt request (OVI) is enabled

Bits 4 and 3—Counter Clear 1 and 0 (CCLR1, CCLR0): These bits select the meth
the timer counter is cleared: by compare-match A or B, or by an external reset input.

Bit 4	Bit 3	
CCLR1	**CCLR0**	**Description**
0	0	Clearing is disabled (
	1	Cleared on compare-match A
1	0	Cleared on compare-match B
	1	Cleared on rising edge of external reset input

Some functions differ between channel 0 and channel 1, because of the cascading funct

	TCR			STCR		
	Bit 2	Bit 1	Bit 0	Bit 1	Bit 0	
Channel	CKS2	CKS1	CKS0	ICKS1	ICKS0	Description
0	0	0	0	—	—	Clock input disabled (Initi
	0	0	1	—	0	ϕ/8 internal clock source, counted on the f
	0	0	1	—	1	ϕ/2 internal clock source, counted on the f
	0	1	0	—	0	ϕ/64 internal clock source, counted on the edge
	0	1	0	—	1	ϕ/32 internal clock source, counted on the edge
	0	1	1	—	0	ϕ/1024 internal clock source, counted on t edge
	0	1	1	—	1	ϕ/256 internal clock source, counted on th edge
	1	0	0	—	—	Counted on TCNT1 overflow signal*
1	0	0	0	—	—	Clock input disabled (Initi
	0	0	1	0	—	ϕ/8 internal clock source, counted on the f
	0	0	1	1	—	ϕ/2 internal clock source, counted on the f
	0	1	0	0	—	ϕ/64 internal clock source, counted on the edge
	0	1	0	1	—	ϕ/128 internal clock source, counted on th edge
	0	1	1	0	—	ϕ/1024 internal clock source, counted on t edge
	0	1	1	1	—	ϕ/2048 internal clock source, counted on t edge
	1	0	0	—	—	Counted on TCNT0 compare-match A*

	1	0	0	—	—	Clock input disabled
Y	0	0	0	—	—	Clock input disabled (Ini
	0	0	1	—	—	φ/4 internal clock source, counted on the
	0	1	0	—	—	φ/256 internal clock source, counted on t edge
	0	1	1	—	—	φ/2048 internal clock source, counted on edge
	1	0	0	—	—	Clock input disabled
Common	1	0	1	—	—	External clock source, counted at rising e
	1	1	0	—	—	External clock source, counted at falling
	1	1	1	—	—	External clock source, counted at both ris falling edges

Note: * If the count input of channel 0 is the TCNT1 overflow signal and that of cha TCNT0 compare-match signal, no incrementing clock will be generated. Do setting.

TCSR1

Bit	7	6	5	4	3	2	1
	CMFB	CMFA	OVF	—	OS3	OS2	OS1
Initial value	0	0	0	1	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	—	R/W	R/W	R/W

TCSRX

Bit	7	6	5	4	3	2	1
	CMFB	CMFA	OVF	ICF	OS3	OS2	OS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W

TCSRY

Bit	7	6	5	4	3	2	1
	CMFB	CMFA	OVF	ICIE	OS3	OS2	OS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W

Note: * Only 0 can be written in bits 7 to 5, and in bit 4 in TCSRX, to clear these flags

TCSR is an 8-bit register that indicates compare-match and overflow statuses (and inpu status in TMRX only), and controls compare-match output.

TCSR0, TCSRX, and TCSRY are initialized to H'00, and TCSR1 is initialized to H'10, and in hardware standby mode.

1	[Setting condition] When TCNT = TCORB

Bit 6—Compare-match Flag A (CMFA): Status flag indicating whether the values o TCORA match.

Bit 6

CMFA	Description
0	[Clearing conditions] (• Read CMFA when CMFA = 1, then write 0 in CMFA • When the DTC is activated by a CMIA interrupt
1	[Setting condition] When TCNT = TCORA

Bit 5 —Timer Overflow Flag (OVF): Status flag indicating that TCNT has overflow from H'FF to H'00).

Bit 5

OVF	Description
0	[Clearing condition] (Read OVF when OVF = 1, then write 0 in OVF
1	[Setting condition] When TCNT overflows from H'FF to H'00

TCSR1

Bit 4—Reserved: This bit cannot be modified and is always read as 1.

TCSRX

Bit 4—Input Capture Flag (ICF): Status flag that indicates detection of a rising edge by a falling edge in the external reset signal after the ICST bit in TCONRI has been set

Bit 4

ICF	Description
0	[Clearing condition] (In Read ICF when ICF = 1, then write 0 in ICF
1	[Setting condition] When a rising edge followed by a falling edge is detected in the external res after the ICST bit in TCONRI has been set to 1

TCSRY

Bit 4—Input Capture Interrupt Enable (ICIE): Selects enabling or disabling of the request by ICF (ICIX) when the ICF bit in TCSRX is set to 1.

Bit 4

ICIE	Description
0	Interrupt request by ICF (ICIX) is disabled (In
1	Interrupt request by ICF (ICIX) is enabled

Timer output is disabled when bits OS3 to OS0 are all 0.

After a reset, the timer output is 0 until the first compare-match occurs.

Bit 3	Bit 2	
OS3	**OS2**	**Description**
0	0	No change when compare-match B occurs (
	1	0 is output when compare-match B occurs
1	0	1 is output when compare-match B occurs
	1	Output is inverted when compare-match B occurs (toggle ou

Bit 1	Bit 0	
OS1	**OS0**	**Description**
0	0	No change when compare-match A occurs (
	1	0 is output when compare-match A occurs
1	0	1 is output when compare-match A occurs
	1	Output is inverted when compare-match A occurs (toggle ou

STCR is an 8-bit readable/writable register that controls register access, the IIC operati
(when the on-chip IIC option is included), and on-chip flash memory (in F-ZTAT versi
also selects the TCNT input clock.

For details on functions not related to the 8-bit timers, see section 3.2.4, Serial Timer C
Register (STCR), and the descriptions of the relevant modules. If a module controlled b
not used, do not write 1 to the corresponding bit.

STCR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 4—I^2C Control (IICS, IICX1, IICX0, IICE): These bits control the bus buf
of port A and the operation of the I^2C bus interface when the IIC option is included on-
section 16.2.7, Serial/Timer Control Register (STCR), for details.

Bit 3—Flash Memory Control Register Enable (FLSHE): Controls CPU access to th
memory control registers, the power-down mode control registers, and the supporting m
control registers. See section 3.2.4, Serial Timer Control Register (STCR), for details..

Bit 2—Reserved: Do not write 1 to this bit.

Bits 1 and 0—Internal Clock Select 1 and 0 (ICKS1, ICKS0): These bits, together w
CKS2 to CKS0 in TCR, select the clock to be input to TCNT. For details, see section 1
Timer Control Register (TCR).

Only bit 1 is described here. For details on functions not related to the 8-bit timers, se
3.2.2 and 5.2.1, System Control Register (SYSCR), and the descriptions of the relevan

Bit 1—Host Interface Enable (HIE): Controls CPU access to 8-bit timer (channel X registers and control registers, and timer connection control registers.

Bit 1

HIE	Description
0	CPU access to 8-bit timer (channel X and Y) data registers and control reg timer connection control registers, is enabled (I
1	CPU access to 8-bit timer (channel X and Y) data registers and control reg timer connection control registers, is disabled

12.2.8 Timer Connection Register S (TCONRS)

Bit	7	6	5	4	3	2	1
	TMRX/Y	ISGENE	HOMOD1	HOMOD0	VOMOD1	VOMOD0	CLMOD1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCONRS is an 8-bit readable/writable register that controls access to the TMRX and T registers and timer connection operation.

TCONRS is initialized to H'00 by a reset and in hardware standby mode.

0 (Initial value)	TMRX	TMRX	TMRX	TMRX	TMRX	TMRX	TMRX
	TCRX	TCSRX	TICRR	TICRF	TCNTX	TCORC	TCORAX
1	TMRY	TMRY	TMRY	TMRY	TMRY	TMRY	
	TCRY	TCSRY	TCORAY	TCORBY	TCNTY	TISR	

12.2.9 Input Capture Register (TICR) [TMRX Additional Function]

Bit	7	6	5	4	3	2	1
Initial value	0	0	0	0	0	0	0
Read/Write	—	—	—	—	—	—	—

TICR is an 8-bit internal register to which the contents of TCNT are transferred on the
of external reset input. The CPU cannot read or write to TICR directly.

The TICR function is used in timer connection. For details, see section 13, Timer Conn

TCORC is an 8-bit readable/writable register. The sum of the contents of TCORC and
continually compared with the value in TCNT. When a match is detected, a compare-
signal is generated. Note, however, that comparison is disabled during the T2 state of
write cycle and a TICR input capture cycle.

TCORC is initialized to H'FF by a reset and in hardware standby mode.

The TCORC function is used in timer connection. For details, see section 13, Timer C

12.2.11 Input Capture Registers R and F (TICRR, TICRF) [TMRX Additional

Bit	7	6	5	4	3	2	1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

TICRR and TICRF are 8-bit read-only registers. When the ICST bit in TCONRI is set
TICRR and TICRF capture the contents of TCNT successively on the rise and fall of t
reset input. When one capture operation ends, the ICST bit is cleared to 0.

TICRR and TICRF are each initialized to H'00 by a reset and in hardware standby mo

The TICRR and TICRF functions are used in timer connection. For details, see section
Input Capture Operation, and section 13, Timer Connection.

TISR is an 8-bit readable/writable register that selects the external clock/reset signal so counter.

TISR is initialized to H'FE by a reset and in hardware standby mode.

Bits 7 to 1—Reserved: Do not write 0 to these bits.

Bit 0—Input Select (IS): Selects the internal synchronization signal (IVG signal) or th clock/reset input pin (VSYNCI/TMIY (TMCIY/TMRIY)) as the external clock/reset si for the counter.

Bit 0

IS	Description
0	IVG signal is selected (H8S/2148 Group) (Ini External clock/reset input is disabled (H8S/2144 Group and H8S/2147N)
1	VSYNCI/TMIY (TMCIY/TMRIY) is selected

MSTPCR comprises two 8-bit readable/writable registers, and is used to perform mod
mode control.

When the MSTP12 bit or MSTP8 bit is set to 1, 8-bit timer operation is halted on char
or channels X and Y, respectively, and a transition is made to module stop mode. For
section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not in
software standby mode.

MSTPCRH Bit 4—Module Stop (MSTP12): Specifies 8-bit timer (channel 0/1) mo
mode.

MSTPCRH Bit 4	
MSTP12	**Description**
0	8-bit timer (channel 0/1) module stop mode is cleared
1	8-bit timer (channel 0/1) module stop mode is set (In

MSTPCRH Bit 0—Module Stop (MSTP8): Specifies 8-bit timer (channel X/Y) and
connection module stop mode.

MSTPCRH Bit 0	
MSTP8	**Description**
0	8-bit timer (channel X/Y) and timer connection module stop mode is clea
1	8-bit timer (channel X/Y) and timer connection module stop mode is set (In

An internal clock created by dividing the system clock (ϕ) can be selected by setting bi
CKS0 in TCR. Figure 12.2 shows the count timing.

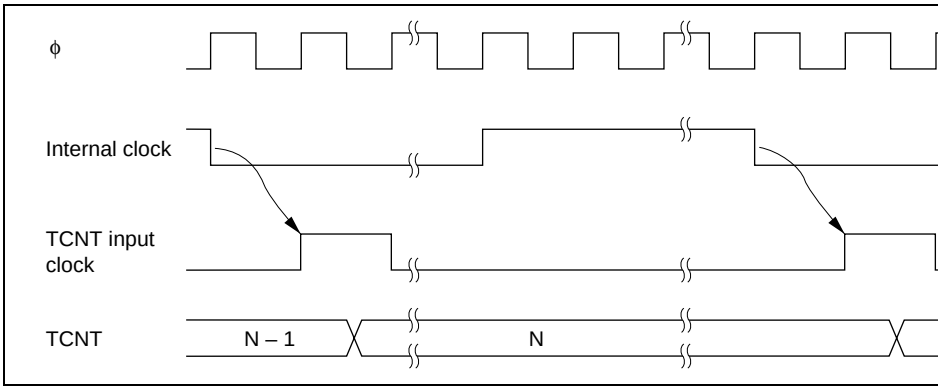


Figure 12.2 Count Timing for Internal Clock Input

External Clock

Three incrementation methods can be selected by setting bits CKS2 to CKS0 in TCR: a
edge, the falling edge, and both rising and falling edges.

Note that the external clock pulse width must be at least 1.5 states for incrementation at
edge, and at least 2.5 states for incrementation at both edges. The counter will not incre
correctly if the pulse width is less than these values.

Figure 12.3 shows the timing of incrementation at both edges of an external clock signa

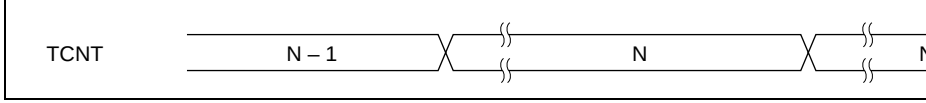


Figure 12.3 Count Timing for External Clock Input

12.3.2 Compare-Match Timing

Setting of Compare-Match Flags A and B (CMFA, CMFB)

The CMFA and CMFB flags in TCSR are set to 1 by a compare-match signal generate
TCOR and TCNT values match. The compare-match signal is generated at the last sta
the match is true, just before the timer counter is updated.

Therefore, when TCOR and TCNT match, the compare-match signal is not generated
next incrementation clock input. Figure 12.4 shows this timing.

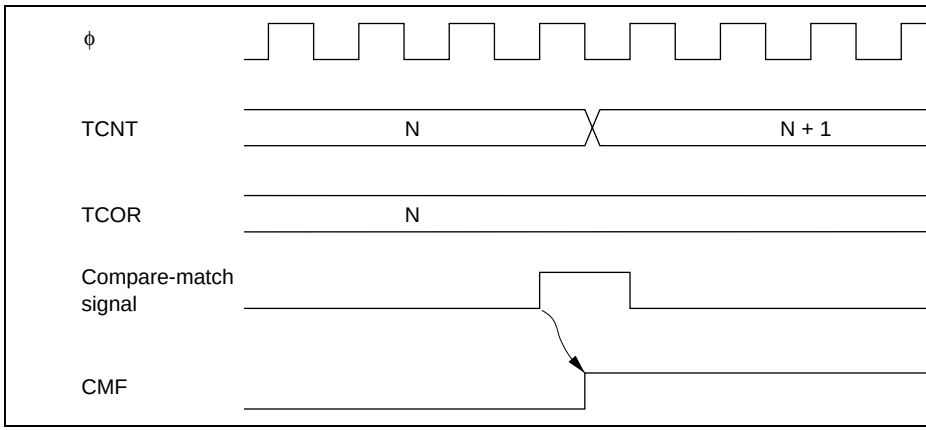


Figure 12.4 Timing of CMF Setting

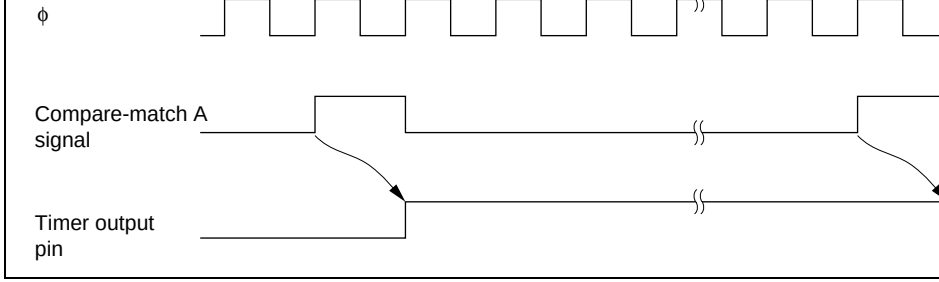


Figure 12.5 Timing of Timer Output

Timing of Compare-Match Clear

TCNT is cleared when compare-match A or B occurs, depending on the setting of the C
CCLR0 bits in TCR. Figure 12.6 shows the timing of this operation.

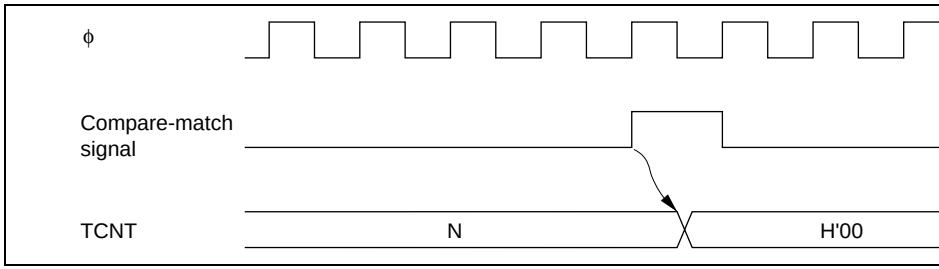


Figure 12.6 Timing of Compare-Match Clear

12.3.3 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
12.7 shows the timing of this operation.

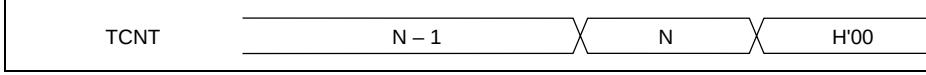


Figure 12.7 Timing of Clearing by External Reset Input

12.3.4 Timing of Overflow Flag (OVF) Setting

OVF in TCSR is set to 1 when the timer count overflows (changes from H'FF to H'00) 12.8 shows the timing of this operation.

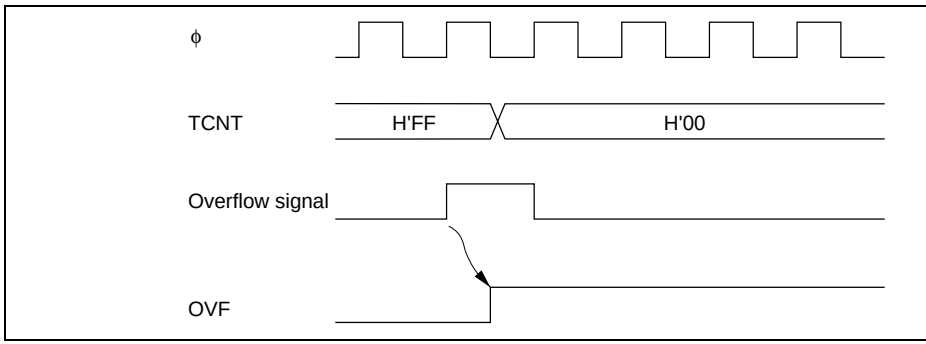


Figure 12.8 Timing of OVF Setting

12.3.5 Operation with Cascaded Connection

If bits CKS2 to CKS0 in either TCR0 or TCR1 are set to B'100, the 8-bit timers of the channels are cascaded. With this configuration, a single 16-bit timer can be used (16-b mode) or compare-matches of 8-bit channel 0 can be counted by the timer of channel match count mode). In this case, the timer operates as described below.

- Counter clear specification
  - If the CCLR1 and CCLR0 bits in TCR0 have been set for counter clear at comp the 16-bit counter (TCNT0 and TCNT1 together) is cleared when a 16-bit comp occurs. The 16-bit counter (TCNT0 and TCNT1 together) is cleared even if cou by the TMRI0 pin has also been set.
  - The settings of the CCLR1 and CCLR0 bits in TCR1 are ignored. The lower 8 b be cleared independently.
- Pin output
  - Control of output from the TMO0 pin by bits OS3 to OS0 in TCSR0 is in accord the 16-bit compare-match conditions.
  - Control of output from the TMO1 pin by bits OS3 to OS0 in TCSR1 is in accord the lower 8-bit compare-match conditions.

Compare-Match Count Mode

When bits CKS2 to CKS0 in TCR1 are B'100, TCNT1 counts compare-match A's for c

Channels 0 and 1 are controlled independently. Conditions such as setting of the CMF generation of interrupts, output from the TMO pin, and counter clearing are in accordan settings for each channel.

Usage Note

If the 16-bit count mode and compare-match count mode are set simultaneously, the inp pulses for TCNT0 and TCNT1 are not generated and thus the counters will stop operati Simultaneous setting of these two modes should therefore be avoided.

(1) Input capture input timing

Figure 12.9 shows the timing of the input capture operation.

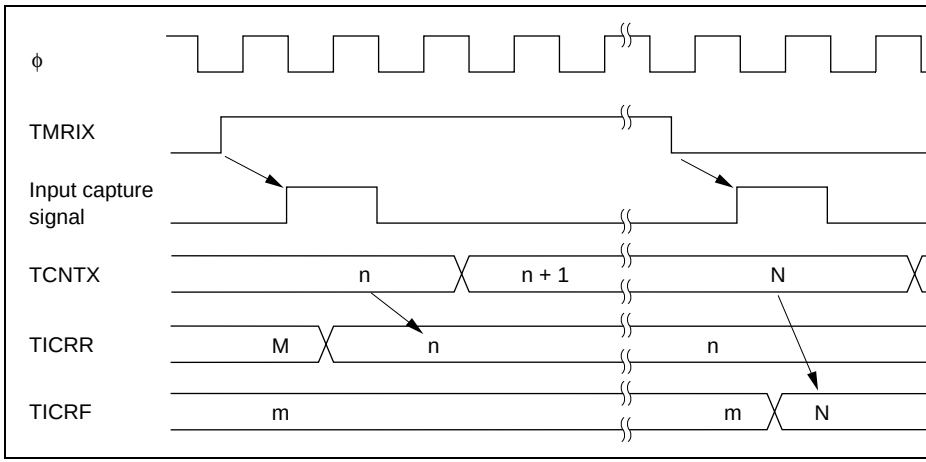


Figure 12.9 Timing of Input Capture Operation

If the input capture signal enters while TICRR and TICRF are being read, it is interna
one system clock (ϕ) period. Figure 12.10 shows the timing of this operation.

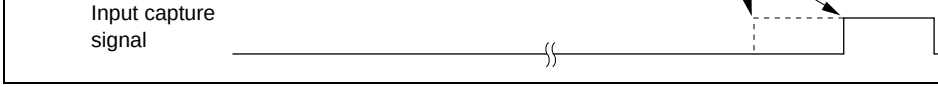


Figure 12.10 Timing of Input Capture Signal
(When Input Capture Input Signal Enters while TICRR and TICRF Are Being

(2) Input capture signal input selection

Input capture input signal (TMRIX) in TMRX is switched by setting bits in the TCONR

Figure 12.11 and Table 12.3 show the input capture signal selections.

See section 13.2.1, Timer Connection Register I (TCONRI), for details.

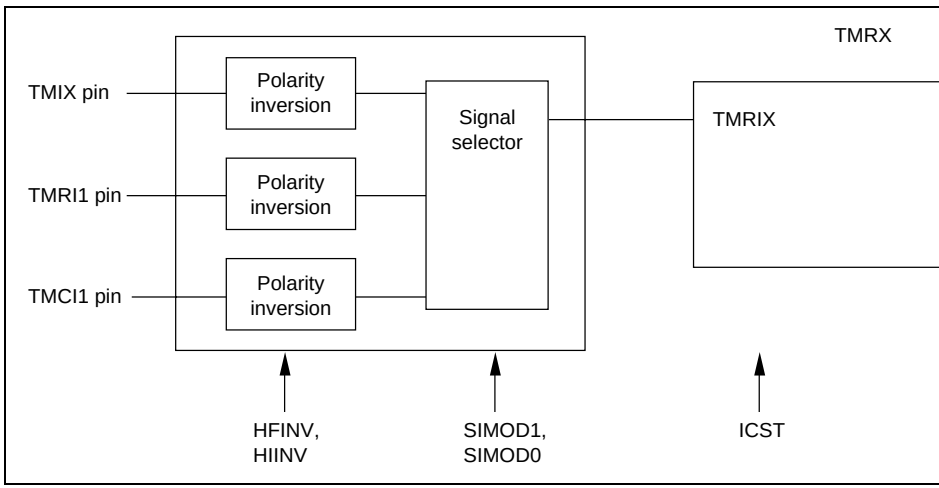


Figure 12.11 Switching of Input Capture Signal

	1	—	0	TMRI1 pin input signal
		—	1	Inverted signal of TMRI1
1	1	—	0	TMCI1 pin input signal
		—	1	Inverted signal of TMCI1

12.4 Interrupt Sources

The TMR0, TMR1, and TMRY 8-bit timers can generate three types of interrupt: com
A and B (CMIA and CMIB), and overflow (OVI). TMRX can generate only an ICIX
interrupt is requested when the corresponding interrupt enable bit is set in TCR or TC
Independent signals are sent to the interrupt controller for each interrupt. It is also pos
activate the DTC by means of CMIA and CMIB interrupts from TMR0, TMR1 and T

An overview of 8-bit timer interrupt sources is given in tables 12.4 to 12.6.

Table 12.4 TMR0 and TMR1 8-Bit Timer Interrupt Sources

Interrupt source	Description	DTC Activation	Interrupt
CMIA	Requested by CMFA	Possible	High
CMIB	Requested by CMFB	Possible	↑
OVI	Requested by OVF	Not possible	Low

Table 12.5 TMRX 8-Bit Timer Interrupt Source

Interrupt source	Description	DTC Activation
ICIX	Requested by ICF	Not possible

12.5 8-Bit Timer Application Example

In the example below, the 8-bit timer is used to generate a pulse output with a selected
as shown in figure 12.12. The control bits are set as follows:

- In TCR, CCLR1 is cleared to 0 and CCLR0 is set to 1 so that the timer counter is cl
TCORA compare-match.
- In TCSR, bits OS3 to OS0 are set to B'0110, causing 1 output at a TCORA compare
0 output at a TCORB compare-match.

With these settings, the 8-bit timer provides output of pulses at a rate determined by TC
a pulse width determined by TCORB. No software intervention is required.

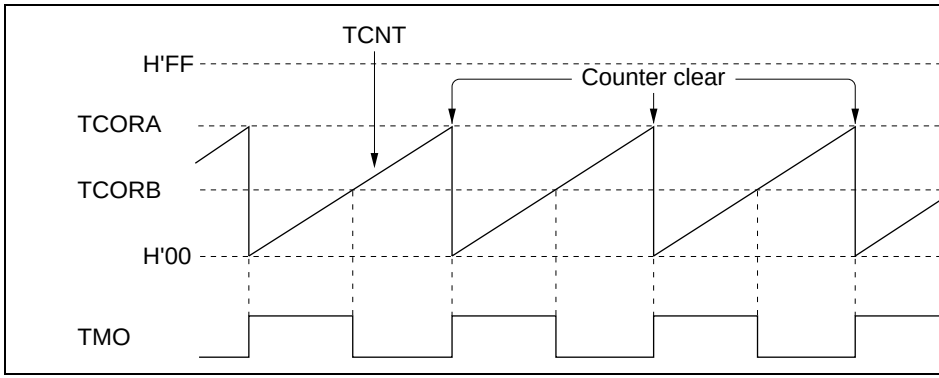


Figure 12.12 Pulse Output (Example)

takes priority, so that the counter is cleared and the write is not performed. Figure 12.1
this operation.

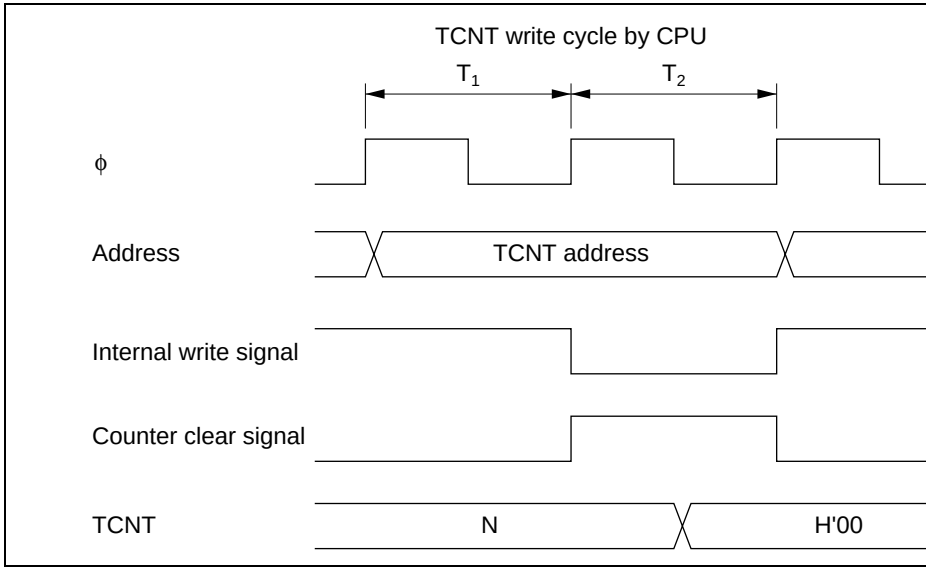


Figure 12.13 Contention between TCNT Write and Clear

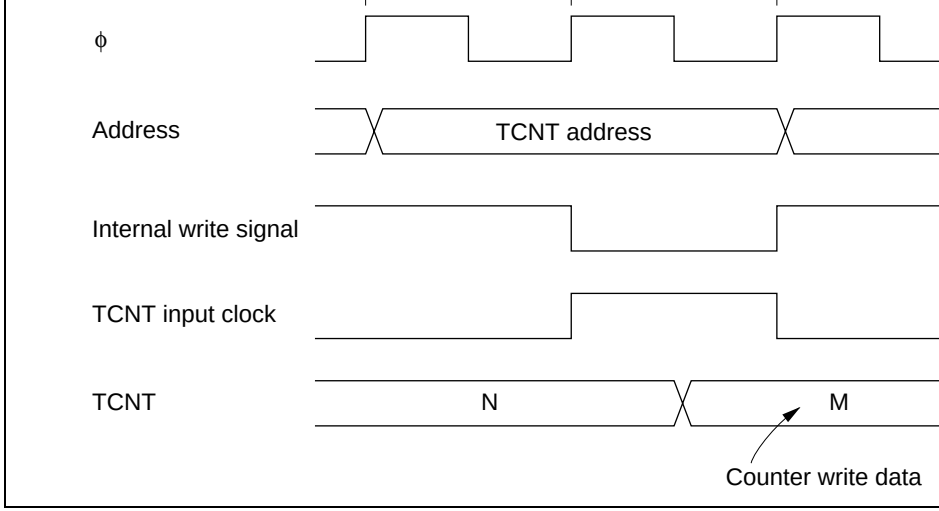


Figure 12.14 Contention between TCNT Write and Increment

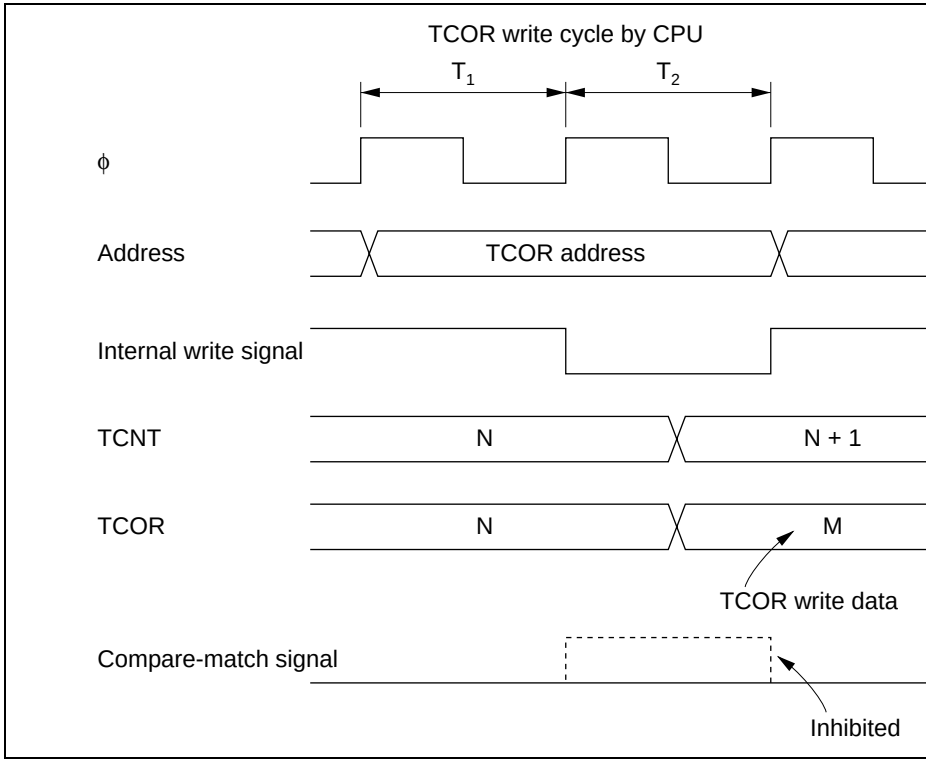


Figure 12.15 Contention between TCOR Write and Compare-Match

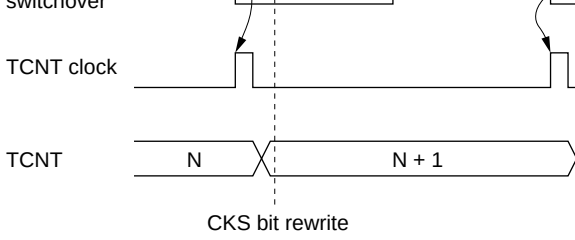
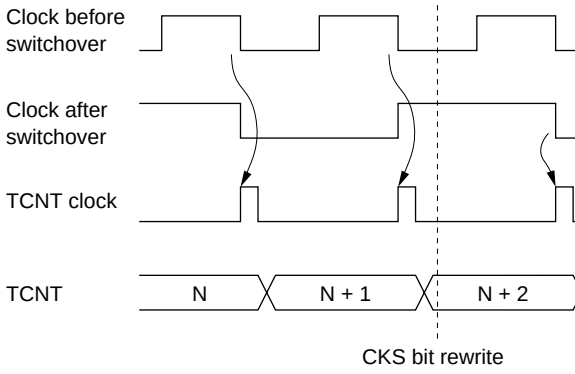
Toggle output	High
1 output	
0 output	
No change	Low

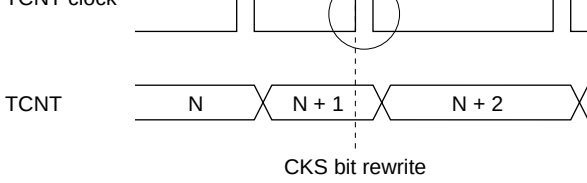
12.6.5 Switching of Internal Clocks and TCNT Operation

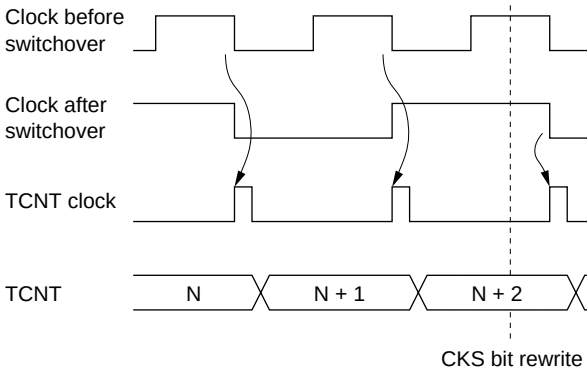
TCNT may increment erroneously when the internal clock is switched over. Table 12.8 relationship between the timing at which the internal clock is switched (by writing to th and CKS0 bits) and the TCNT operation

When the TCNT clock is generated from an internal clock, the falling edge of the intern pulse is detected. If clock switching causes a change from high to low level, as shown i table 12.8, a TCNT clock pulse is generated on the assumption that the switchover is a edge. This increments TCNT.

Erroneous incrementation can also happen when switching between internal and extern

		
2	Switching from low to high*2	



4	Switching from high to high	

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.

input/output of the single-free-running timer (FRT) channel and the three 8-bit timer c
(TMR1, TMRX, and TMRY). This capability can be used to implement complex func
PWM decoding and clamp waveform output. All the timers are initially set for indepe
operation.

13.1.1 Features

The features of the timer connection facility are as follows.

- Five input pins and four output pins, all of which can be designated for phase inve
  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 d
- TMRX can be used for PWM input signal decoding and clamp waveform generati
- An external clock signal divided by TMR1 can be used as the FRT capture input s
- An internal synchronization signal can be generated using the FRT and TMRY.
- A signal generated/modified using an input signal and timer connection can be sel
  output.

13.1.2 Block Diagram

Figure 13.1 shows a block diagram of the timer connection facility.

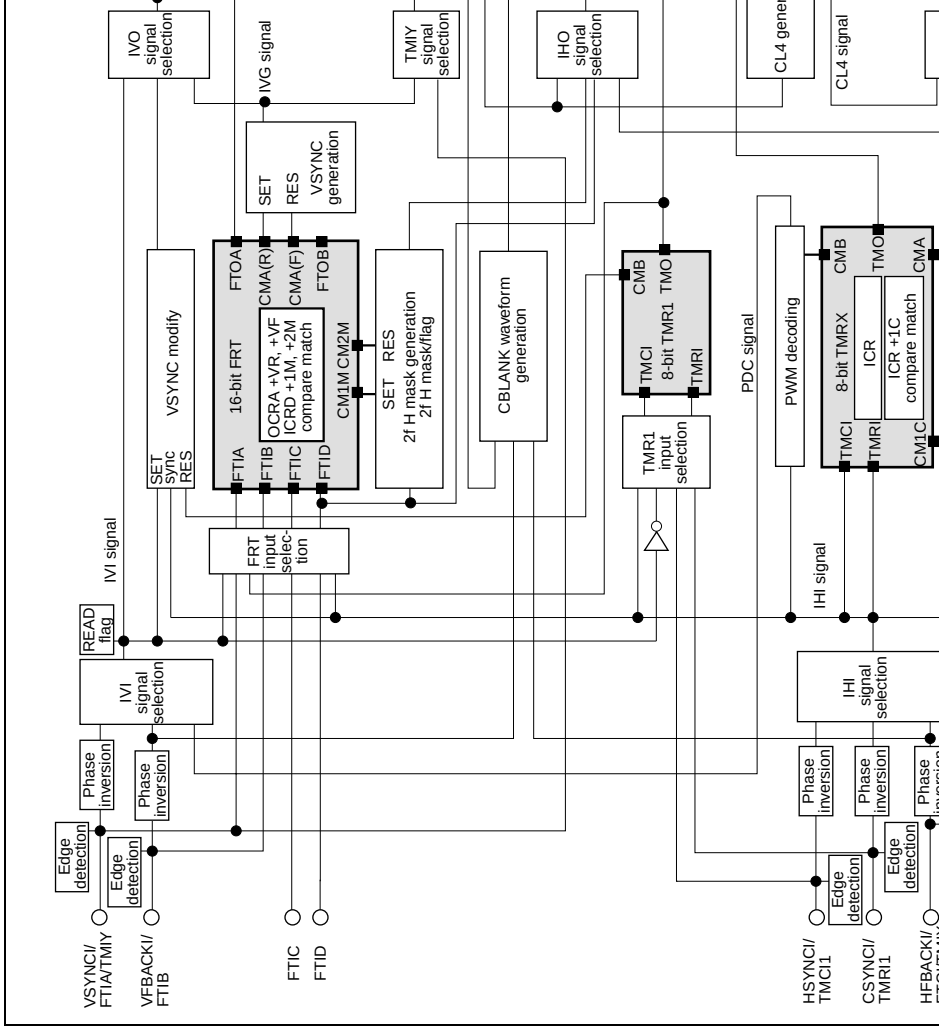


Figure 13.1 Block Diagram of Timer Connection Facility

signal input pin			input pin or FTIA inpu input pin
Horizontal synchronization signal input pin	HSYNCI	Input	Horizontal synchroniz input pin or TMCI1 inp
Composite synchronization signal input pin	CSYNCI	Input	Composite synchroni input pin or TMRI1 inp
Spare vertical synchronization signal input pin	VFBACKI	Input	Spare vertical synchr signal input pin or FT
Spare horizontal synchronization signal input pin	HFBACKI	Input	Spare horizontal sync signal input pin or FT pin/TMIX input pin
Vertical synchronization signal output pin	VSYNCO	Output	Vertical synchronizati output pin or FTOA o
Horizontal synchronization signal output pin	HSYNCO	Output	Horizontal synchroniz output pin or TMO1 o
Clamp waveform output pin	CLAMPO	Output	Clamp waveform outp FTIC input pin
Blanking waveform output pin	CBLANK	Output	Blanking waveform ou

Timer connection register O	TCONRO	R/W	H'00	H'F
Timer connection register S	TCONRS	R/W	H'00	H'F
Edge sense register	SEDGR	R/(W)*2	H'00*3	H'F
Module stop control register	MSTPRH	R/W	H'3F	H'F
	MSTPRL	R/W	H'FF	H'F

Notes: 1. Lower 16 bits of the address.
2. Bits 7 to 2: Only 0 can be written to clear the flags.
3. Bits 1 and 0: Undefined (reflect the pin states).

13.2 Register Descriptions

13.2.1 Timer Connection Register I (TCONRI)

Bit	7	6	5	4	3	2	1	
	SIMOD1	SIMOD0	SCONE	ICST	HFINV	VFINV	HIINV	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TCONRI is an 8-bit readable/writable register that controls connection between timers, source for synchronization signal input, phase inversion, etc.

TCONR1 is initialized to H'00 by a reset and in hardware standby mode.

1	Separate mode	HSYNCI input	VSY

Bit 5—Synchronization Signal Connection Enable (SCONE): Selects the signal so FRT FTI input and the TMR1 TMCI1/TMRI1 input.

Bit 5	Description					
SCONE	**Mode**	**FTIA**	**FTIB**	**FTIC**	**FTID**	**TM**
0	Normal connection (Initial value)	FTIA input	FTIB input	FTIC input	FTID input	TM inpu
1	Synchronization signal connection mode	IVI signal	TMO1 signal	VFBACKI input	IHI signal	IHI sign

Bit 4—Input Capture Start Bit (ICST): The TMRX external reset input (TMRIX) is to the IHI signal. TMRX has input capture registers (TICR, TICRR, and TICRF). TIC TICRF can measure the width of a short pulse by means of a single capture operation control of the ICST bit. When a rising edge followed by a falling edge is detected on T the ICST bit is set to 1, the contents of TCNT at those points are captured into TICRR respectively, and the ICST bit is cleared to 0.

Bit 4	
ICST	**Description**
0	The TICRR and TICRF input capture functions are stopped (I [Clearing condition] When a rising edge followed by a falling edge is detected on TMRIX
1	The TICRR and TICRF input capture functions are operating (Waiting for detection of a rising edge followed by a falling edge on TMRIX [Setting condition] When 1 is written in ICST after reading ICST = 0

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

Bit 2

VFINV	Description
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

Bit 1

HIINV	Description
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 CSYNCI inputs

Bit 0

VIINV	Description
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

etc.

TCONRO is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 and 4—Output Enable (HOE, VOE, CLOE, CBOE): These bits control enabling/disabling of horizontal synchronization signal (HSYNCO), vertical synchron signal (VSYNCO), clamp waveform (CLAMPO), and blanking waveform (CBLANK When output is disabled, the state of the relevant pin is determined by the port DR and TMR, and PWM settings.

Output enabling/disabling control does not affect the port, FRT, or TMR input functio FRT and TMR input signal sources are determined by the SCONE bit in TCONRI.

Bit 7

HOE	Description	
0	The P44/TMO1/HIRQ1/HSYNCO pin functions as the P44/TMO1/ HIRQ1 pin	(
1	The P44/TMO1/HIRQ1/HSYNCO pin functions as the HSYNCO pin	

Bit 6

VOE	Description	
0	The P61/FTOA/$\overline{\text{KIN1}}$/CIN1/VSYNCO pin functions as the P61/FTOA/ $\overline{\text{KIN1}}$/CIN1 pin	(
1	The P61/FTOA/$\overline{\text{KIN1}}$/CIN1/VSYNCO pin functions as the VSYNCO pin	

CBOE	Description	
0	The P27/A15/PW15/CBLANK pin functions as the P27/A15/PW15 pin	(In
1	In mode 1 (expanded mode with on-chip ROM disabled): The P27/A15/PW15/CBLANK pin functions as the A15 pin In modes 2 and 3 (modes with on-chip ROM enabled): The P27/A15/PW15/CBLANK pin functions as the CBLANK pin	

Bits 3 to 0—Output Synchronization Signal Inversion (HOINV, VOINV, CLOINV CBOINV): These bits select inversion of the output phase of the horizontal synchroniz (HSYNCO), the vertical synchronization signal (VSYNCO), the clamp waveform (CLA and the blank waveform (CBLANK).

Bit 3

HOINV	Description	
0	The IHO signal is used directly as the HSYNCO output	(In
1	The IHO signal is inverted before use as the HSYNCO output	

Bit 2

VOINV	Description	
0	The IVO signal is used directly as the VSYNCO output	(In
1	The IVO signal is inverted before use as the VSYNCO output	

Bit 1

CLOINV	Description	
0	The CLO signal (CL1, CL2, CL3, or CL4 signal) is used directly as the CLAMPO output	(In
1	The CLO signal (CL1, CL2, CL3, or CL4 signal) is inverted before use as the CLAMPO output	

Bit	7	6	5	4	3	2	1
	TMRX/Y	ISGENE	HOMOD1	HOMOD0	VOMOD1	VOMOD0	CLMOD1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCONRS is an 8-bit readable/writable register that selects 8-bit timer TMRX/TMRY the synchronization signal output signal source and generation method.

TCONRS is initialized to H'00 by a reset and in hardware standby mode.

Bit 7—TMRX/TMRY Access Select (TMRX/Y): The TMRX and TMRY registers accessed when the HIE bit in SYSCR is cleared to 0. In the H8S/2148 Group, some of registers and the TMRY registers are assigned to the same memory space addresses (H H'FFF5), and the TMRX/Y bit determines which registers are accessed. In the H8S/21 and H8S/2147N, there is no control of TMRY register access by this bit.

Bit 7

TMRX/Y	Description
0	The TMRX registers are accessed at addresses H'FFF0 to H'FFF5 (
1	The TMRY registers are accessed at addresses H'FFF0 to H'FFF5

Bit 6—Internal Synchronization Signal Select (ISGENE): Selects internal synchro signals (IHG, IVG, and CL4 signals) as the signal sources for the IHO, IVO, and CLC

	1	0	The CL1 signal is selected
		1	
1	0	0	The IHG signal is selected
		1	
	1	0	
		1	

Bits 3 and 2—Vertical Synchronization Output Mode Select 1 and 0 (VOMOD1, V

These bits select the signal source and generation method for the IVO signal.

Bit 6	Bit 3	Bit 2	
ISGENE	**VOMOD1**	**VOMOD0**	**Description**
0	0	0	The IVI signal (without fall modification or IHI synchronization) is selected (Ini
		1	The IVI signal (without fall modification, with IHI synchronization) is selected
	1	0	The IVI signal (with fall modification, without IHI synchronization) is selected
		1	The IVI signal (with fall modification and IHI synchronization) is selected
1	0	0	The IVG signal is selected
		1	
	1	0	
		1	

		1	
1	0	0	The CL4 signal is selected
		1	
	1	0	
		1	

13.2.4 Edge Sense Register (SEDGR)

Bit	7	6	5	4	3	2	1
	VEDG	HEDG	CEDG	HFEDG	VFEDG	PREQF	IHI
Initial value	0	0	0	0	0	0	—*2
Read/Write	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R

Notes: 1. Only 0 can be written, to clear the flags.
2. The initial value is undefined since it depends on the pin states.

SEDGR is an 8-bit readable/writable register used to detect a rising edge on the timer input pins and the occurrence of 2fH modification, and to determine the phase of the I signals.

The upper 6 bits of SEDGR are initialized to 0 by a reset and in hardware standby mo initial value of the lower 2 bits is undefined, since it depends on the pin states.

Bit 6—HSYNCI Edge (HEDG): Detects a rising edge on the HSYNCI pin.

Bit 6

HEDG	Description
0	[Clearing condition] (Ir When 0 is written in HEDG after reading HEDG = 1
1	[Setting condition] When a rising edge is detected on the HSYNCI pin

Bit 5—CSYNCI Edge (CEDG): Detects a rising edge on the CSYNCI pin.

Bit 5

CEDG	Description
0	[Clearing condition] (Ir When 0 is written in CEDG after reading CEDG = 1
1	[Setting condition] When a rising edge is detected on the CSYNCI pin

Bit 4—HFBACKI Edge (HFEDG): Detects a rising edge on the HFBACKI pin.

Bit 4

HFEDG	Description
0	[Clearing condition] (Ir When 0 is written in HFEDG after reading HFEDG = 1
1	[Setting condition] When a rising edge is detected on the HFBACKI pin

Bit 2—Pre-Equalization Flag (PREQF): Detects the occurrence of an IHI signal 2f
modification condition. The generation of a falling/rising edge in the IHI signal during
interval is expressed as the occurrence of a 2fH modification condition. For details, se
13.3.4, IHI Signal 2fH Modification.

Bit 2

PREQF	Description
0	[Clearing condition] (When 0 is written in PREQF after reading PREQF = 1
1	[Setting condition] When an IHI signal 2fH modification condition is detected

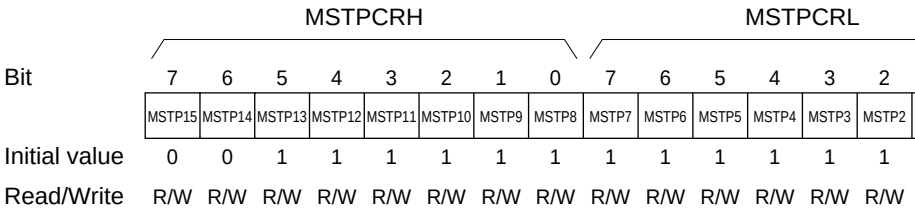
Bit 1—IHI Signal Level (IHI): Indicates the current level of the IHI signal. Signal so
phase inversion selection for the IHI signal depends on the contents of TCONRI. Read
determine whether the input signal is positive or negative, then maintain the IHI signa
phase by modifying TCONRI.

Bit 1

IHI	Description
0	The IHI signal is low
1	The IHI signal is high

1	The IVI signal is high	

13.2.5 Module Stop Control Register (MSTPCR)



MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mod
When the MSTP13, MSTP12, and MSTP8 bits are set to 1, the 16-bit free-running time
timer channels 0 and 1, and 8-bit timer channels X and Y and timer connection, respect
and enter module stop mode. See section 25.5., Module Stop Mode, for details.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini
software standby mode.

MSTPCRH Bit 5—Module Stop (MSTP13): Specifies FRT module stop mode.

MSTPCRH Bit 5	
MSTP13	**Description**
0	FRT module stop mode is cleared
1	FRT module stop mode is set (In

MSTPCRH Bit 0—Module Stop (MSTP8): Specifies 8-bit timer channel X and Y a connection module stop mode.

MSTPCRH Bit 0	
MSTP8	**Description**
0	8-bit timer channel X and Y and timer connection module stop mode is cle
1	8-bit timer channel X and Y and timer connection module stop mode is (set

13.3 Operation

13.3.1 PWM Decoding (PDC Signal Generation)

The timer connection facility and TMRX can be used to decode a PWM signal in whi are represented by the pulse width. To do this, a signal in which a rising edge is gener regular intervals must be selected as the IHI signal.

The timer counter (TCNT) in TMRX is set to count the internal clock pulses and to be the rising edge of the external reset signal (IHI signal). The value to be used as the thr deciding the pulse width is written in TCORB. The PWM decoder contains a delay lat uses the IHI signal as data and compare-match signal B (CMB) as a clock, and the sta signal (the result of the pulse width decision) at the compare-match signal B timing af reset by the rise of the IHI signal is output as the PDC signal. The pulse width setting TICRR and TICRF of TMRX can be used to determine the pulse width decision thres Examples of TCR and TCORB settings are shown in tables 13.3 and 13.4, and the tim shown in figure 13.2.

2 to 0	CKS2 to CKS0	001	Incremented on internal clock: φ

Table 13.4 Examples of TCORB (Pulse Width Threshold) Settings

	φ:10 MHz	φ: 12 MHz	φ: 16 MHz	φ: 20 MH
H'07	0.8 µs	0.67 µs	0.5 µs	0.4 µs
H'0F	1.6 µs	1.33 µs	1 µs	0.8 µs
H'1F	3.2 µs	2.67 µs	2 µs	1.6 µs
H'3F	6.4 µs	5.33 µs	4 µs	3.2 µs
H'7F	12.8 µs	10.67 µs	8 µs	6.4 µs

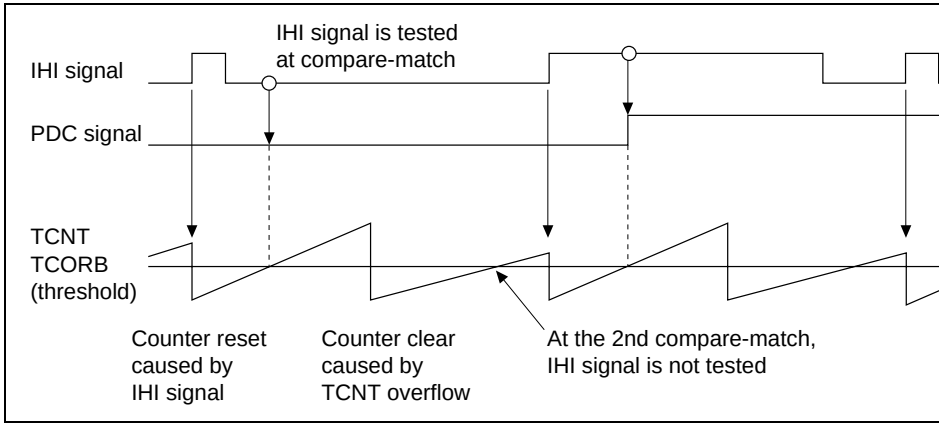


Figure 13.2 Timing Chart for PWM Decoding

and the CL2 signal can be specified by TCORA.

The rise of the CL3 signal can be specified as simultaneous with the sampling of the f signal using the system clock, and the fall of the CL3 signal can be specified by TCOR signal can also fall when the IHI signal rises.

TCNT in TMRX is set to count internal clock pulses and to be cleared on the rising ed external reset signal (IHI signal).

The value to be used as the CL1 signal pulse width is written in TCORA. Write a valu more in TCORA when internal clock ϕ is selected as the TMRX counter clock, and a H'01 or more when $\phi/2$ is selected. When internal clock ϕ is selected, the CL1 signal p is (TCORA set value +3 ±0.5). When the CL2 signal is used, the setting must be made 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. The TICR re TMRX captures the value of TCNT at the inverse of the external reset signal edge (in falling edge of the IHI signal). The timing of the fall of the CL3 signal is determined b the contents of TICR and TCORC. Caution is required if the rising edge of the IHI sig the fall timing set by the contents of TCORC, since the IHI signal will cause the CL3

Examples of TMRX TCR settings are the same as those in table 13.3. The clamp wave charts are shown in figures 13.3 and 13.4.

Since the rise of the CL1 and CL2 signals is synchronized with the edge of the IHI sig their fall is synchronized with the system clock, the pulse width variation is equivalen resolution of the system clock.

Both the rise and the fall of the CL3 signal are synchronized with the system clock an width is fixed, but there is a variation in the phase relationship with the IHI signal equ the resolution of the system clock.

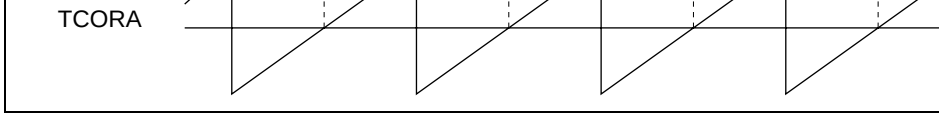


Figure 13.3 Timing Chart for Clamp Waveform Generation (CL1 and CL2 S

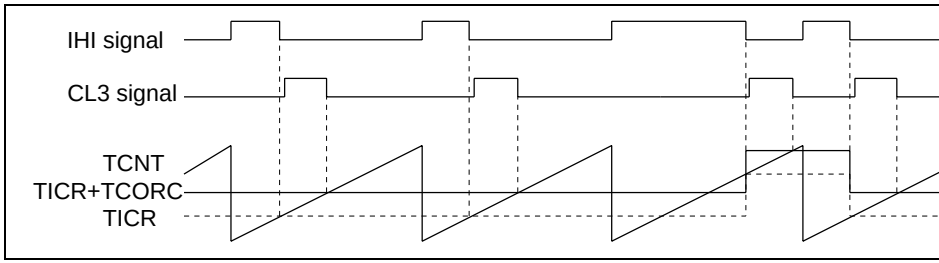


Figure 13.4 Timing Chart for Clamp Waveform Generation (CL3 Signa

13.3.3 Measurement of 8-Bit Timer Divided Waveform Period

The timer connection facility, TMR1, and the free-running timer (FRT) can be used to
period of an IHI signal divided waveform. Since TMR1 can be cleared by a rising edge
IVI signal, the rise and fall of the IHI signal divided waveform can be virtually synchro
the IVI signal. This enables period measurement to be carried out efficiently.

To measure the period of an IHI signal divided waveform, TCNT in TMR1 is set to cou
external clock (IHI signal) pulses and to be cleared on the rising edge of the external re
(inverted IVI signal). The value to be used as the division factor is written in TCORA,
TMO output method is specified by the OS bits in TCSR. Examples of TCR and TCSR
are shown in table 13.5, and the timing chart for measurement of the IVI signal and IHI
divided waveform periods is shown in figure 13.5. The period of the IHI signal divided
is given by (ICRD(3) – ICRD(2)) × the resolution.

				IVI signal)
	2 to 0	CKS2 to CKS0	101	TCNT is incremented on edge of the external cloc signal)
TCSR in TMR1	3 to 0	OS3 to OS0	0011	Not changed by compare output inverted by compa (toggle output): division b
				or
			1001	when TCORB < TCORA on compare-match B, an on compare-match A: div 256
TCR in FRT	6	IEDGB	0/1	0: FRC value is transferr on falling edge of inpu input B (IHI divided sig waveform)
				1: FRC value is transferr on rising edge of input input B (IHI divided sig waveform)
	1 and 0	CKS1, CKS0	01	FRC is incremented on i clock: ϕ/8
TCSR in FRT	0	CCLRA	0	FRC clearing is disabled

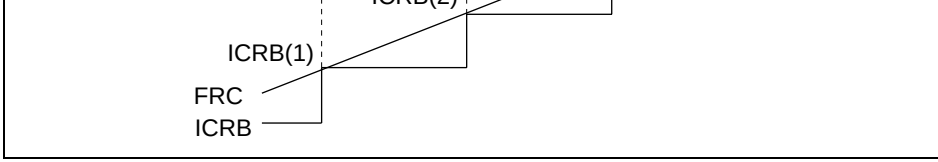


Figure 13.5 Timing Chart for Measurement of IVI Signal and IHI Signal Divided Waveform Periods

13.3.4 IHI Signal and 2fH Modification

By using the timer connection FRT, even if there is a part of the IHI signal with twice t frequency, this can be eliminated. In order for this function to operate properly, the dut 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 inte between the two compare-matches is called a mask interval. A value equivalent to appr 1/3 the IHI signal period is written in OCRDM. ICRD is set so that capture is performe rise of the IHI signal.

Since the IHI signal supplied to the IHO signal selection circuit is normally set on the r IHI signal and reset on the fall, its waveform is the same as that of the original IHI sign 2fH modification is selected, IHI signal edge detection is disabled during mask interval is also disabled during these intervals.

Examples of FRT TCR settings are shown in table 13.6, and the 2fH modification timi shown in figure 13.6.

TCOR in FRT	7	ICRDMS	1	ICRD is set to the operat which OCRDM is used
OCRDM in FRT	7 to 0	OCRDM7 to 0	H'01 to H'FF	Specifies the period duri ICRD operation is maske

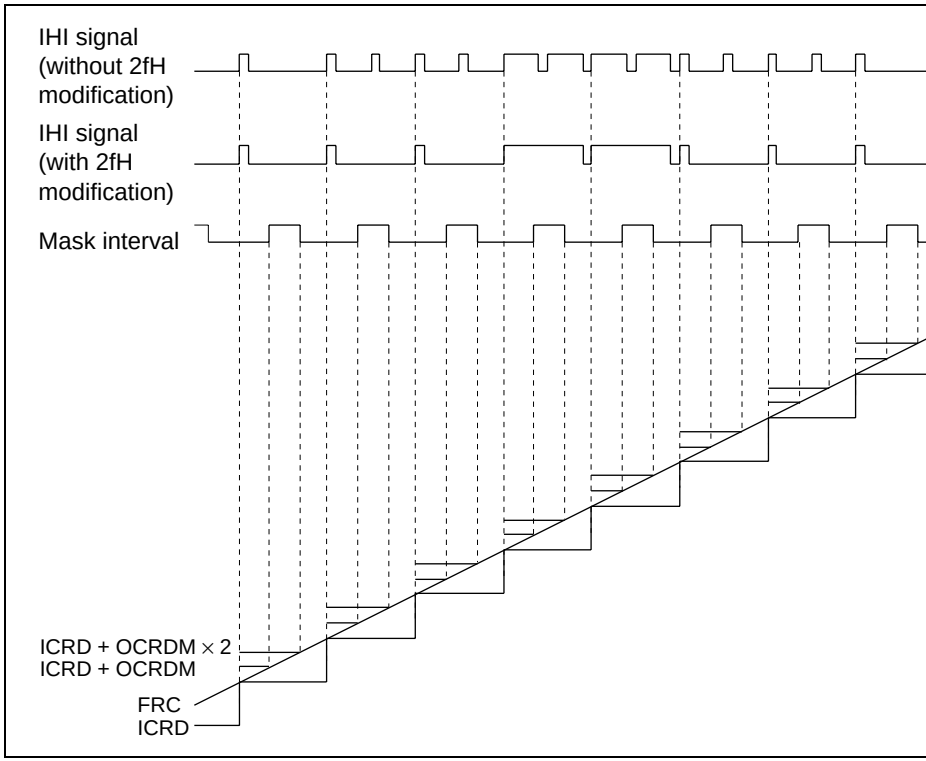


Figure 13.6 2fH Modification Timing Chart

written in TCORB.

Since the IVI signal supplied to the IVO signal selection circuit is normally set on the r
IVI signal and reset on the fall, its waveform is the same as that of the original IVI sign
fall modification is selected, a reset is performed on a TMR1 TCORB compare-match.

The fall of the waveform generated in this way can be synchronized with the rise of the
regardless of whether or not fall modification is selected.

Examples of TMR1 TCORB, TCR, and TCSR settings are shown in table 13.7, and the
modification/IHI synchronization timing chart is shown in figure 13.7.

				signal)
	2 to 0	CKS2 to CKS0	101	TCNT is incremented on the r the external clock (IHI signal)
TCSR in TMR1	3 to 0	OS3 to OS0	0011	Not changed by compare-mat inverted by compare-match A output)
				or
			1001	when TCORB ≤ TCORA, 1 ou compare-match B, 0 output or match A
TOCRB in TMR1			H'03 (example)	Compare-match on the 4th (ex of the IHI signal after the rise inverse of the IVI signal

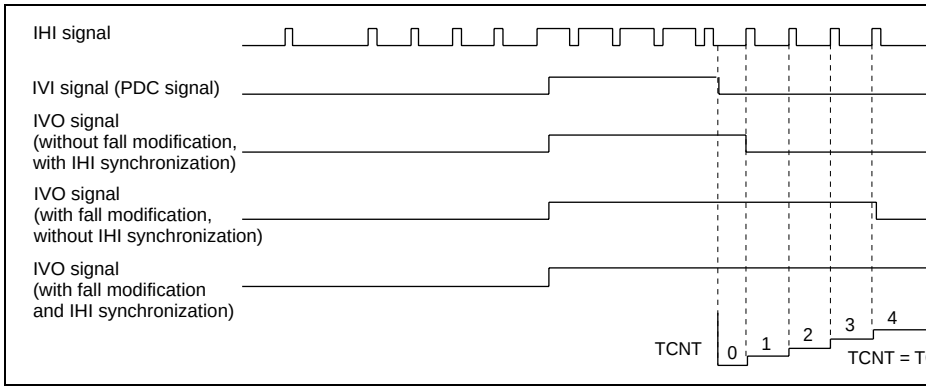


Figure 13.7 Fall Modification/IHI Synchronization Timing Chart

OCRAR or OCRAF, alternately, each time a compare-match occurs. A value correspon
0 interval of the IVG signal is written in OCRAR, and a value corresponding to the 1 in
the IVG signal is written in OCRAF. The IVG signal is set by a compare-match after an
addition, and reset by a compare-match after an OCRAF addition.

The IHG signal is the TMRY 8-bit timer output. TMRY is set to count internal clock pu
to be cleared on TCORA compare-match, to fix the period and set the timer output. TC
so as to reset the timer output. The IVG signal is connected as the TMRY reset input (T
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
signal, and has a 1 interval of 6 system clock periods.

Examples of settings of TCORA, TCORB, TCR, and TCSR in TMRY, and OCRAR, C
and TCR in the FRT, are shown in table 13.8, and the IHG signal/IVG signal timing ch
shown in figure 13.8.

	4 and 3	CCLR1, CCLR0	01	TCNT is cleared by compare-	
	2 to 0	CKS2 to CKS0	001	TCNT is incremented on inter $\phi/4$	
TCSR in TMRY	3 to 0	OS3 to OS0	0110	0 output on compare-match B 1 output on compare-match A	
TOCRA in TMRY			H'3F (example)	IHG signal period = $\phi \times 256$	
TOCRB in TMRY			H'03 (example)	IHG signal 1 interval = $\phi \times 16$	
TCR in FRT	1 and 0	CKS1, CKS0	01	FRC is incremented on interna	
OCRAR in FRT			H'7FEF (example)	IVG signal 0 interval = $\phi \times 262016$	IVG sign $\phi \times 2621$ 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 r which OCRAR and OCRAF ar	

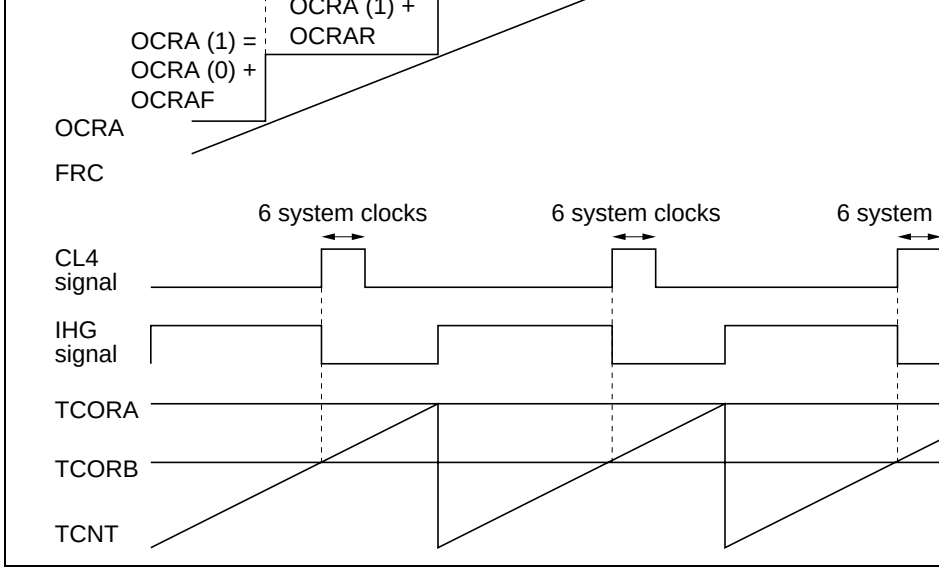


Figure 13.8 IVG Signal/IHG Signal/CL4 Signal Timing Chart

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 b
		IHG signal	Internal synchronization signal is outpu
S-on-G mode	CSYNCI input	IHI signal (without 2fH modification)	CSYNCI input (composite synchroniza is output directly
		IHI signal (with 2fH modification)	Double-frequency part of CSYNCI inpu synchronization signal) is eliminated be
		CL1 signal	CSYNCI input (composite synchroniza horizontal synchronization signal part i before output
		IHG signal	Internal synchronization signal is outpu
Composite mode	HSYNCI input	IHI signal (without 2fH modification)	HSYNCI input (composite synchroniza is output directly
		IHI signal (with 2fH modification)	Double-frequency part of HSYNCI inpu synchronization signal) is eliminated be
		CL1 signal	HSYNCI input (composite synchroniza horizontal synchronization signal part i before output
		IHG signal	Internal synchronization signal is outpu
Separate mode	HSYNCI input	IHI signal (without 2fH modification)	HSYNCI input (horizontal synchronizat output directly
		IHI signal (with 2fH modification)	Meaningless unless there is a double-f part in the HSYNCI input (horizontal synchronization signal)
		CL1 signal	HSYNCI input (horizontal synchronizat interval is changed before output
		IHG signal	Internal synchronization signal is outpu

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 inpu synchronized with HFBACKI input
		IVI signal (with fall modification, without IHI synchronization)	VFBACKI input fall is modified befo
		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 o
S-on-G mode or composite mode	PDC signal	IVI signal (without fall modification or IHI synchronization)	CSYNCI/HSYNCI input (composite synchronization signal) vertical synchronization signal part is sepa before output
		IVI signal (without fall modification, with IHI synchronization)	CSYNCI/HSYNCI input (composite synchronization signal) vertical synchronization signal part is sepa signal is synchronized with CSYNC input before output
		IVI signal (with fall modification, without IHI synchronization)	CSYNCI/HSYNCI input (composite synchronization signal) vertical synchronization signal part is sepa fall is modified before output
		IVI signal (with fall modification and IHI synchronization)	CSYNCI/HSYNCI input (composite synchronization signal) vertical synchronization signal part is sepa modified, and signal is synchronize CSYNCI/HSYNCI input before outp
		IVG signal	Internal synchronization signal is o

modification, without IHI synchronization)	fall is modified before output
IVI signal (with fall modification and IHI synchronization)	VSYNCI input (vertical synchroniz fall is modified and signal is synch HSYNCI input (horizontal synchro signal) before output
IVG signal	Internal synchronization signal is

13.3.9 CBLANK Output

Using the signals generated/selected with timer connection, it is possible to generate a 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 TCONF IVO signal.

The composition logic is shown in figure 13.9.

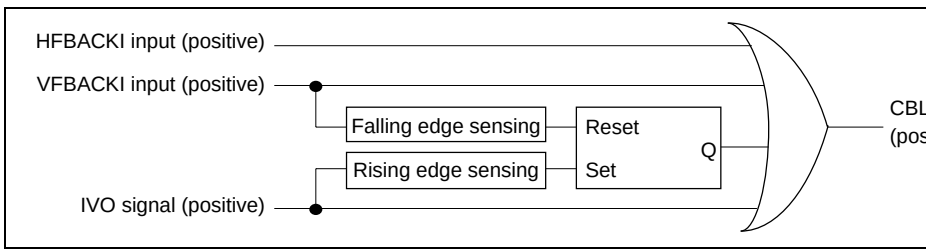


Figure 13.9 CBLANK Output Waveform Generation

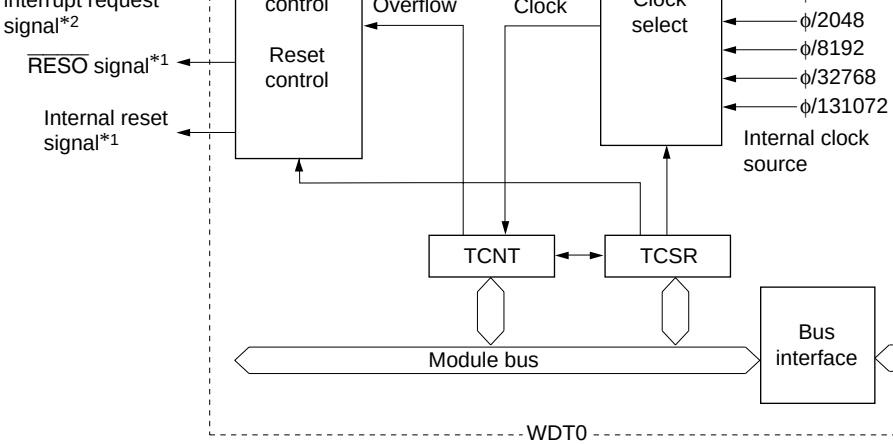
also generate an internal reset signal or internal NMI interrupt signal.

When this watchdog function is not needed, the WDT can be used as an interval timer timer mode, an interval timer interrupt is generated each time the counter overflows.

14.1.1 Features

WDT features are listed below.

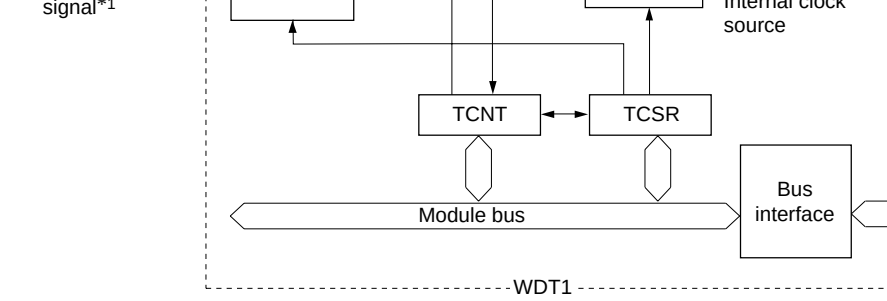
- Switchable between watchdog timer mode and interval timer mode
- Internal reset or internal interrupt generated when the timer counter overflows
  - WOVI interrupt generation in interval timer mode
  - Choice of internal reset or NMI interrupt generation in watchdog timer mode
- $\overline{\text{RESO}}$ output in watchdog timer mode
  - In watchdog timer mode, a low-level signal is output from the $\overline{\text{RESO}}$ pin when overflows (when internal reset is selected)
- Choice of 8 (WDT0) or 16 (WDT1) counter input clocks
  - Maximum WDT interval: system clock period × 131072 × 256
  - Subclock can be selected for the WDT1 input counter
    Maximum interval when the subclock is selected: subclock period × 256 × 256



Legend:
TCSR: Timer control/status register
TCNT: Timer counter

Notes: 1. RESO pin output goes low when the internal reset signal is generated by overflow o
in either WDT0 or WDT1. The reset of the WDT that overflowed first takes precede
the internal reset signal.
2. The internal NMI interrupt request signal can be output independently by either WD
WDT1. The interrupt controller does not distinguish between NMI interrupt requests
WDT0 and WDT1.

Figure 14.1 (a) Block Diagram of WDT0



Legend:
TCSR: Timer control/status register
TCNT: Timer counter

Notes: 1. $\overline{\text{RESO}}$ pin output goes low when the internal reset signal is generated by overflow of
in either WDT0 or WDT1. The reset of the WDT that overflowed first takes precedenc
the internal reset signal.
2. The internal NMI interrupt request signal can be output independently by either WDT
WDT1. The interrupt controller does not distinguish between NMI interrupt requests fr
WDT0 and WDT1.

Figure 14.1 (b) Block Diagram of WDT1

14.1.3 Pin Configuration

Table 14.1 describes the WDT input pin.

Table 14.1 WDT Pin

Name	Symbol	I/O	Function
Reset output pin	$\overline{\text{RESO}}$	Output	Watchdog timer mode counter ov output
External subclock input pin	EXCL	Input	WDT1 prescaler counter input clo

0	Timer control/status register 0	TCSR0	R/(W)*3	H'00	H'FFA8
	Timer counter 0	TCNT0	R/W	H'00	H'FFA8
1	Timer control/status register 1	TCSR1	R/(W)*3	H'00	H'FFEA
	Timer counter 1	TCNT1	R/W	H'00	H'FFEA
Common	System control register	SYSCR	R/W	H'09	H'FFC4

Notes: 1. Lower 16 bits of the address.
2. For details of write operations, see section 14.2.4, Notes on Register Access
3. Only 0 can be written in bit 7, to clear the flag.

14.2 Register Descriptions

14.2.1 Timer Counter (TCNT)

Bit	7	6	5	4	3	2	1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TCNT is an 8-bit readable/writable* up-counter.

When the TME bit is set to 1 in TCSR, TCNT starts counting pulses generated from the clock source selected by bits CKS2 to CKS0 in TCSR. When the count overflows (chan H'FF to H'00), the OVF flag in TCSR is set to 1. Watchdog timer overflow signal ($\overline{\text{RES}}$ an internal reset, NMI interrupt, interval timer interrupt (WOVI), etc., can be generated on the mode selected by the WT/$\overline{\text{IT}}$ bit and RST/$\overline{\text{NMI}}$ bit.

- TCSR0

Bit	7	6	5	4	3	2	1
	OVF	WT/$\overline{IT}$	TME	RSTS	RST/$\overline{NMI}$	CKS2	CKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Only 0 can be written, to clear the flag.

- TCSR1

Bit	7	6	5	4	3	2	1
	OVF	WT/$\overline{IT}$	TME	PSS	RST/$\overline{NMI}$	CKS2	CKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Only 0 can be written, to clear the flag.

TCSR is an 8-bit readable/writable* register. Its functions include selecting the clock s
input to TCNT, and the timer mode.

TCR is initialized to H'00 by a reset and in hardware standby mode. It is not initialized
standby mode.

Note: * TCSR is write-protected by a password to prevent accidental overwriting. F
see section 14.2.4, Notes on Register Access.

1	[Setting condition] When TCNT overflows (changes from H'FF to H'00) (When internal reset request generation is selected in watchdog timer mod cleared automatically by the internal reset.)

Note: * When OVF is polled and the interval timer interrupt is disabled, OVF = 1 mus at least twice.

Bit 6—Timer Mode Select (WT/$\overline{IT}$): Selects whether the WDT is used as a watchdog interval timer. If used as an interval timer, the WDT generates an interval timer interru (WOVI) when TCNT overflows. If used as a watchdog timer, the WDT generates a res interrupt when TCNT overflows. When internal reset is selected in watchdog timer mo level signal is output from the $\overline{RESO}$ pin.

Bit 6

WT/$\overline{IT}$	Description
0	Interval timer mode: Sends the CPU an interval timer interrupt request (WO TCNT overflows (Ini
1	Watchdog timer mode: Generates a reset or NMI interrupt when TCNT ove At the same time, a low-level signal is output from the $\overline{RESO}$ pin (when inte is selected)

Bit 5—Timer Enable (TME): Selects whether TCNT runs or is halted.

Bit 5

TME	Description
0	TCNT is initialized to H'00 and halted (Ini
1	TCNT counts

1	TCNT counts φSUB-based prescaler (PSS) divided clock pulses

Bit 3—Reset or NMI (RST/$\overline{\text{NMI}}$): Specifies whether an internal reset or NMI interru requested on TCNT overflow in watchdog timer mode.

Bit 3

RST/$\overline{\text{NMI}}$	Description
0	An NMI interrupt is requested
1	An internal reset is requested

Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS0): These bits select an internal clock obtained by dividing the system clock (φ), or subclock (φSUB) for input to TCNT.

- WDT0 input clock selection

Bit 2	Bit 1	Bit 0	Description	
CKS2	CKS1	CKS0	Clock	Overflow Period* (when φ = 20
0	0	0	φ/2 (Initial value)	25.6 µs
		1	φ/64	819.2 µs
	1	0	φ/128	1.6 ms
		1	φ/512	6.6 ms
1	0	0	φ/2048	26.2 ms
		1	φ/8192	104.9 ms
	1	0	φ/32768	419.4 ms
		1	φ/131072	1.68 s

Note: * The overflow period is the time from when TCNT starts counting up from H' overflow occurs.

			1	ϕ/512	6.6 ms
	1	0	0	ϕ/2048	26.2 ms
			1	ϕ/8192	104.9 ms
		1	0	ϕ/32768	419.4 ms
			1	ϕ/131072	1.68 s
1	0	0	0	ϕSUB/2	15.6 ms
			1	ϕSUB/4	31.3 ms
		1	0	ϕSUB/8	62.5 ms
			1	ϕSUB/16	125 ms
	1	0	0	ϕSUB/32	250 ms
			1	ϕSUB/64	500 ms
		1	0	ϕSUB/128	1 s
			1	ϕSUB/256	2 s

Note: * The overflow period is the time from when TCNT starts counting up from H'0 overflow occurs.

14.2.3 System Control Register (SYSCR)

Bit	7	6	5	4	3	2	1
	CS2E	IOSE	INTM1	INTM0	XRST	NMIEG	HIE
Initial value	0	0	0	0	1	0	0
Read/Write	R/W	R/W	R	R/W	R	R/W	R/W

Only bit 3 is described here. For details on functions not related to the watchdog timer, sections 3.2.2 and 5.2.1, System Control Register (SYSCR), and the descriptions of the modules.

Bit 3—External Reset (XRST): Indicates the reset source. When the watchdog timer i reset can be generated by watchdog timer overflow in addition to external reset input. X

14.2.4 Notes on Register Access

The watchdog timer's TCNT and TCSR registers differ from other registers in being r
to write to. The procedures for writing to and reading these registers are given below.

Writing to TCNT and TCSR (Example of WDT0)

These registers must be written to by a word transfer instruction. They cannot be writt
byte transfer instructions.

Figure 14.2 shows the format of data written to TCNT and TCSR. TCNT and TCSR b
same write address. For a write to TCNT, the upper byte of the written word must con
and the lower byte must contain the write data. For a write to TCSR, the upper byte of
word must contain H'A5 and the lower byte must contain the write data. This transfers
data from the lower byte to TCNT or TCSR.

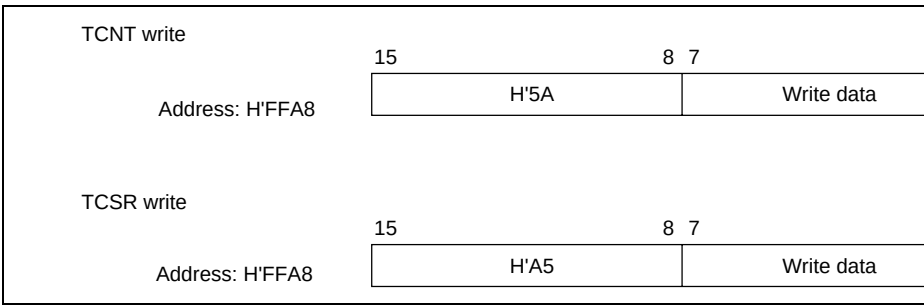


Figure 14.2 Format of Data Written to TCNT and TCSR (Example of W

Reading TCNT and TCSR (Example of WDT0)

These registers are read in the same way as other registers. The read addresses are H'F
TCSR, and H'FFA9 for TCNT.

internal reset or NMI interrupt request is generated.

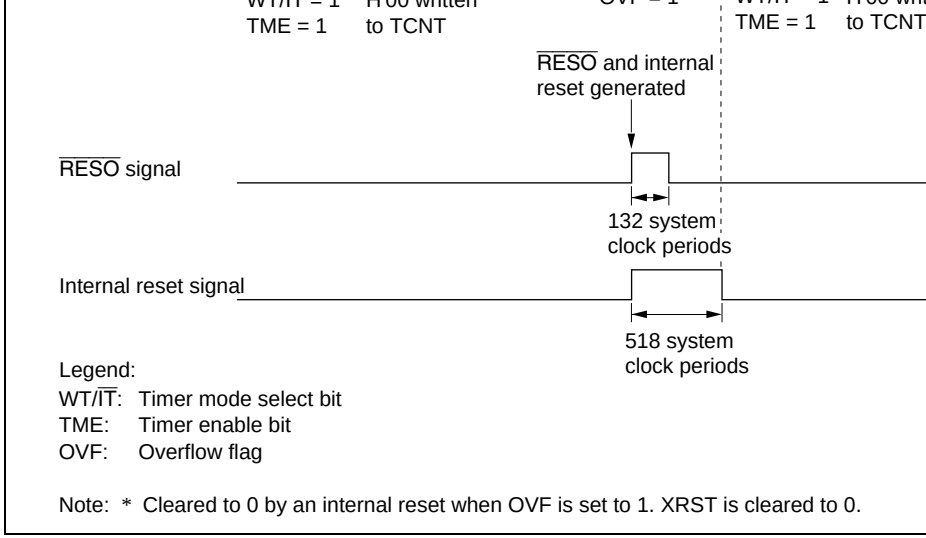
When the RST/$\overline{\text{NMI}}$ bit is set to 1, the chip is reset for 518 system clock periods (518 ϕ
counter overflow, and at the same time a low-level signal is output from the $\overline{\text{RESO}}$ pin
states. This is illustrated in figure 14.3. The system can be reset using this $\overline{\text{RESO}}$ signal

When the RST/$\overline{\text{NMI}}$ bit cleared to 0, an NMI interrupt request is generated by a counte
In this case, the $\overline{\text{RESO}}$ output signal remains high.

An internal reset request from the watchdog timer and reset input from the $\overline{\text{RES}}$ pin are
via the same vector. The reset source can be identified from the value of the XRST bit i

If a reset caused by an input signal from the $\overline{\text{RES}}$ pin and a reset caused by WDT overf
simultaneously, 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 N
handled via the same vector. Simultaneous handling of a watchdog timer NMI interrup
and an NMI pin interrupt request must therefore be avoided.



Legend:
WT/$\overline{IT}$: Timer mode select bit
TME: Timer enable bit
OVF: Overflow flag

Note: * Cleared to 0 by an internal reset when OVF is set to 1. XRST is cleared to 0.

Figure 14.3 Operation in Watchdog Timer Mode (RST/$\overline{NMI}$ = 1)

14.3.2 Interval Timer Operation

To use the WDT as an interval timer, clear the WT/$\overline{IT}$ bit in TCSR to 0 and set the TM
An interval timer interrupt (WOVI) is generated each time TCNT overflows, provided
WDT is operating as an interval timer, as shown in figure 14.4. This function can be u
generate interrupt requests at regular intervals.

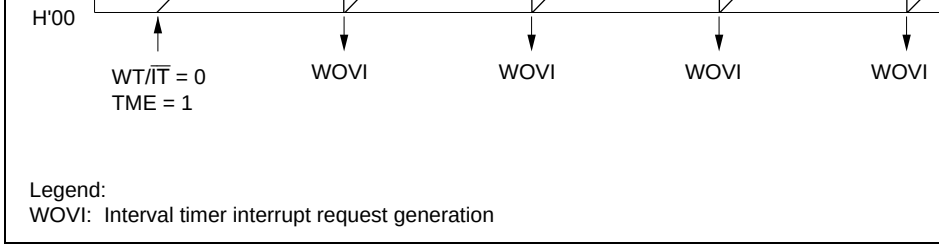


Figure 14.4 Operation in Interval Timer Mode

14.3.3 Timing of Setting of Overflow Flag (OVF)

The OVF flag is set to 1 if TCNT overflows during interval timer operation. At the sam
interval timer interrupt (WOVI) is requested. This timing is shown in figure 14.5.

If NMI request generation is selected in watchdog timer mode, when TCNT overflows
bit in TCSR is set to 1 and at the same time an NMI interrupt is requested.

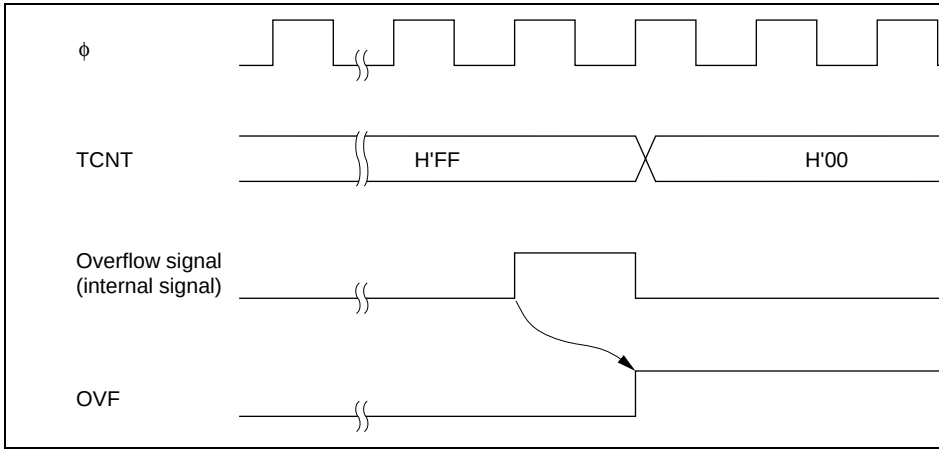


Figure 14.5 Timing of OVF Setting

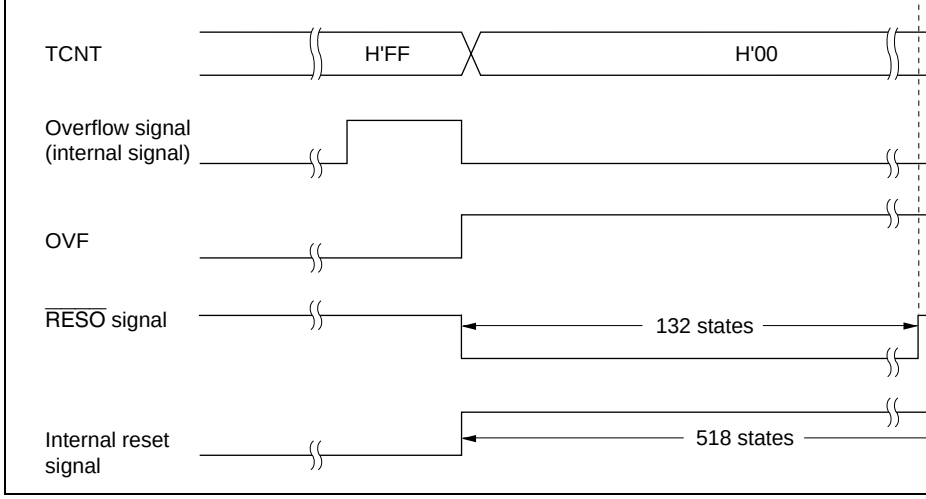


Figure 14.6 $\overline{\text{RESO}}$ Signal Output Timing

14.4 Interrupts

During interval timer mode operation, an overflow generates an interval timer interrup
The interval timer interrupt is requested whenever the OVF flag is set to 1 in TCSR. C
cleared to 0 in the interrupt handling routine. When NMI interrupt request generation
watchdog timer mode, an overflow generates an NMI interrupt request.

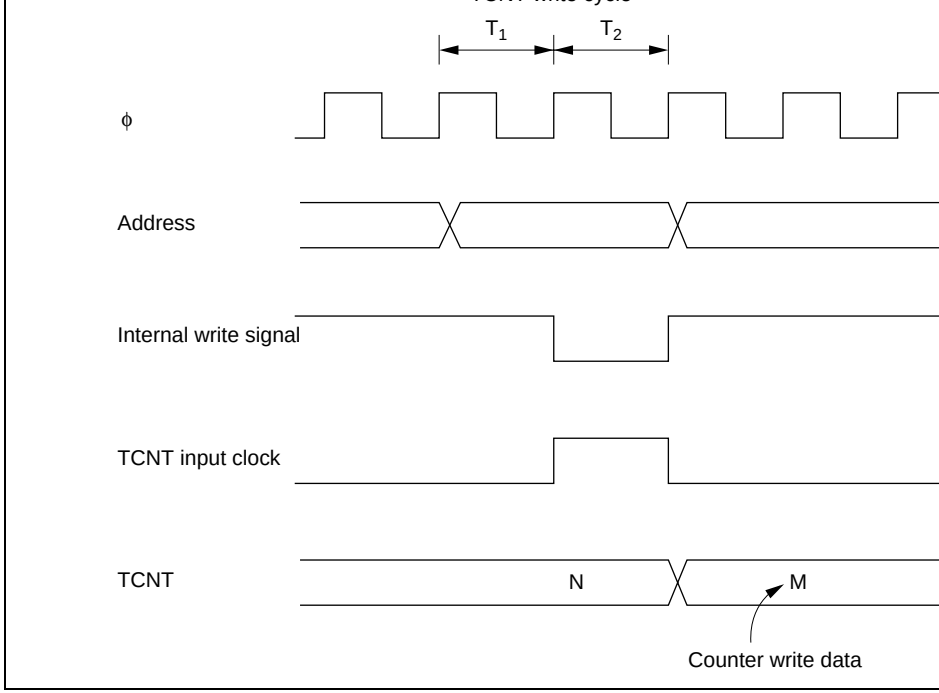


Figure 14.7 Contention between TCNT Write and Increment

14.5.2 Changing Value of CKS2 to CKS0

If bits CKS2 to CKS0 in TCSR are written to while the WDT is operating, errors could the incrementation. Software must stop the watchdog timer (by clearing the TME bit to changing the value of bits CKS2 to CKS0.

If the $\overline{\text{RESO}}$ output signal is input to the chip's $\overline{\text{RES}}$ pin, the chip will not be initialize
Ensure that the $\overline{\text{RESO}}$ signal is not logically input to the chip's $\overline{\text{RES}}$ pin. When resetti
system with the $\overline{\text{RESO}}$ signal, use a circuit such as that shown in figure 14.8.

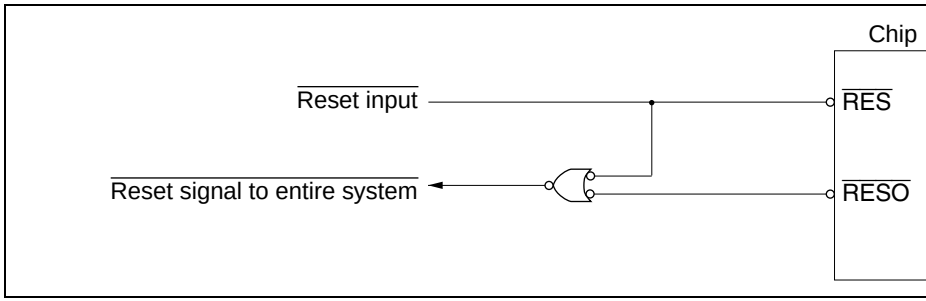


Figure 14.8 Sample Circuit for System Reset by $\overline{\text{RESO}}$ Signal

14.5.5 Counter Value in Transitions between High-Speed Mode, Subactive Mo Watch Mode

If the mode is switched between high-speed mode and subactive mode or between hig
mode and watch mode when WDT1 is used as a realtime clock counter, an error will o
counter value when the internal clock is switched.

When the mode is switched from high-speed mode to subactive mode or watch mode,
increment timing is delayed by approximately 2 or 3 clock cycles when the WDT1 co
switched from the main clock to the subclock.

Also, since the main clock oscillator is halted during subclock operation, when the mo
switched from watch mode or subactive mode to high-speed mode, the clock is not su
internal oscillation stabilizes. As a result, after oscillation is started, counter incremen
during the oscillation stabilization time set by bits STS2 to STS0 in SBYCR, and there
corresponding discrepancy in the counter value.

flag set and TCSR read is occurred, OVF = 1 is read but OVF can not be cleared by wri
to OVF.

In this case, reading TCSR when OVF = 1 two times meet the requirements of OVF cle
condition. Please read TCSR when OVF = 1 two times before writing with 0 to OVF.

```
LOOP    BTST.B   #7, @TCSR       ; OVF flag read
        BEQ      LOOP            ; if OVF=1, exit from loop
        MOV.B    @TCSR, R0L      ; OVF=1 read again
        MOV.W    #H'A521, R0     ; OVF flag clear
        MOV.W    R0,@TCSR        ;  :
```

One of the three SCI channels can transmit and receive IrDA communication wavefor
IrDA specification version 1.0.

15.1.1 Features

SCI features are listed below.

- Choice of asynchronous or synchronous serial communication mode

  Asynchronous mode:

  — Serial data communication is executed using an asynchronous system in which synchronization is achieved character by character

    Serial data communication can be carried out with standard asynchronous com chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asy Communication Interface Adapter (ACIA)

  — A multiprocessor communication function is provided that enables serial data communication with a number of processors

  — Choice of 12 serial data transfer formats

Data length:	7 or 8 bits
Stop bit length:	1 or 2 bits
Parity:	Even, odd, or none
Multiprocessor bit:	1 or 0

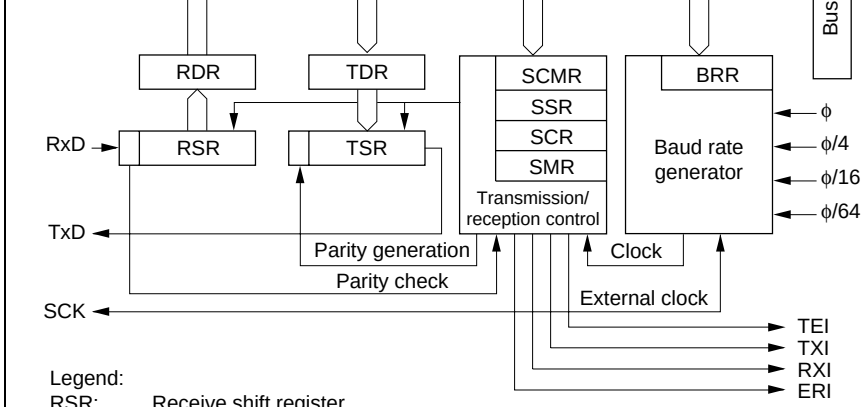
  — Receive error detection: Parity, overrun, and framing errors

  — Break detection: Break can be detected by reading the RxD pin level directly in case of a framing error

- Full-duplex communication capability
  - The transmitter and receiver are mutually independent, enabling transmission an to be executed simultaneously
  - Double-buffering is used in both the transmitter and the receiver, enabling conti transmission and continuous reception of serial data
- LSB-first or MSB-first transfer can be selected
  - This selection can be made regardless of the communication mode (with the exc 7-bit data transfer in asynchronous mode)*

Note: * LSB-first transfer is used in the examples in this section.

- Built-in baud rate generator allows any bit rate to be selected
- Choice of serial clock source: internal clock from baud rate generator or external cl SCK pin
- Capability of transmit and receive clock output
  - The P86/SCK1 and P42/SCK2 pins are CMOS type outputs
  - The P52/SCK0 pin is an NMOS push-pull type output in the H8S/2148 Group a H8S/2147N, and is a CMOS output in the H8S/2144 Group (When the P52/SC used as an output in the H8S/2148 Group and H8S/2147N, external pull-up resis connected in order to output high level)
- The transmit-data-empty interrupt and receive-data-full interrupt can activate the da controller (DTC) to execute data transfer
- Four interrupt sources
  - Four interrupt sources (transmit-data-empty, transmit-end, receive-data-full, and error) that can issue requests independently
  - The transmit-data-empty interrupt and receive-data-full interrupt can activate th transfer controller (DTC) to execute data transfer



Legend:

RSR:	Receive shift register
RDR:	Receive data register
TSR:	Transmit shift register
TDR:	Transmit data register
SMR:	Serial mode register
SCR:	Serial control register
SSR:	Serial status register
SCMR:	Serial interface mode register
BRR:	Bit rate register

Figure 15.1 Block Diagram of SCI

	Transmit data pin 0	TxD0	Output	SCI0 transmit data outpu
1	Serial clock pin 1	SCK1	I/O	SCI1 clock input/output
	Receive data pin 1	RxD1	Input	SCI1 receive data input
	Transmit data pin 1	TxD1	Output	SCI1 transmit data outpu
2	Serial clock pin 2	SCK2	I/O	SCI2 clock input/output
	Receive data pin 2	RxD2/IrRxD	Input	SCI2 receive data input (normal/IrDA)
	Transmit data pin 2	TxD2/IrTxD	Output	SCI2 transmit data outpu (normal/IrDA)

Note: * The abbreviations SCK, RxD, and TxD are used in the text, omitting the cha number.

15.1.4 Register Configuration

The SCI has the internal registers shown in table 15.2. These registers are used to speci asynchronous mode or synchronous mode, the data format, and the bit rate, and to cont transmitter/receiver.

	Receive data register 0	RDR0	R	H'00
	Serial interface mode register 0	SCMR0	R/W	H'F2
1	Serial mode register 1	SMR1	R/W	H'00
	Bit rate register 1	BRR1	R/W	H'FF
	Serial control register 1	SCR1	R/W	H'00
	Transmit data register 1	TDR1	R/W	H'FF
	Serial status register 1	SSR1	R/(W)*2	H'84
	Receive data register 1	RDR1	R	H'00
	Serial interface mode register 1	SCMR1	R/W	H'F2
2	Serial mode register 2	SMR2	R/W	H'00
	Bit rate register 2	BRR2	R/W	H'FF
	Serial control register 2	SCR2	R/W	H'00
	Transmit data register 2	TDR2	R/W	H'FF
	Serial status register 2	SSR2	R/(W)*2	H'84
	Receive data register 2	RDR2	R	H'00
	Serial interface mode register 2	SCMR2	R/W	H'F2
	Keyboard comparator control register	KBCOMP	R/W	H'00
Common	Module stop control register	MSTPCRH	R/W	H'3F
		MSTPCRL	R/W	H'FF

Notes: 1. Lower 16 bits of the address.

2. Only 0 can be written, to clear flags.

3. Some serial communication interface registers are assigned to the same ad other registers. In this case, register selection is performed by the IICE bit i timer control register (STCR).

RSR is a register used to receive serial data.

The SCI sets serial data input from the RxD pin in RSR in the order received, starting w
LSB (bit 0), and converts it to parallel data. When one byte of data has been received, i
transferred to RDR automatically.

RSR cannot be directly read or written to by the CPU.

15.2.2 Receive Data Register (RDR)

Bit	7	6	5	4	3	2	1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R	R	R	R	R	R	R	

RDR is a register that stores received serial data.

When the SCI has received one byte of serial data, it transfers the received serial data f
RDR where it is stored, and completes the receive operation. After this, RSR is receive

Since RSR and RDR function as a double buffer in this way, continuous receive operat
performed.

RDR is a read-only register, and cannot be written to by the CPU.

RDR is initialized to H'00 by a reset, and in standby mode, watch mode, subactive mod
mode, and module stop mode.

To perform serial data transmission, the SCI first transfers transmit data from TDR to
sends the data to the TxD pin starting with the LSB (bit 0).

When transmission of one byte is completed, the next transmit data is transferred from
TSR, and transmission started, automatically. However, data transfer from TDR to TS
performed if the TDRE bit in SSR is set to 1.

TSR cannot be directly read or written to by the CPU.

15.2.4 Transmit Data Register (TDR)

Bit	7	6	5	4	3	2	1
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TDR is an 8-bit register that stores data for serial transmission.

When the SCI detects that TSR is empty, it transfers the transmit data written in TDR
starts serial transmission. Continuous serial transmission can be carried out by writing
transmit data to TDR during serial transmission of the data in TSR.

TDR can be read or written to by the CPU at all times.

TDR is initialized to H'FF by a reset, and in standby mode, watch mode, subactive mo
mode, and module stop mode.

generator clock source.

SMR can be read or written to by the CPU at all times.

SMR is initialized to H'00 by a reset, and in standby mode, watch mode, subactive mod
mode, and module stop mode.

Bit 7—Communication Mode (C/$\overline{A}$): Selects asynchronous mode or synchronous mo
SCI operating mode.

Bit 7

C/$\overline{A}$	Description	
0	Asynchronous mode	(
1	Synchronous mode	

Bit 6—Character Length (CHR): Selects 7 or 8 bits as the data length in asynchronou
synchronous mode, a fixed data length of 8 bits is used regardless of the CHR setting.

Bit 6

CHR	Description	
0	8-bit data	(
1	7-bit data*	

Note: * When 7-bit data is selected, the MSB (bit 7) of TDR is not transmitted, and L
first/MSB-first selection is not available.

1	Parity bit addition and checking enabled*

Note: * When the PE bit is set to 1, the parity (even or odd) specified by the O/$\overline{E}$ bit transmit data before transmission. In reception, the parity bit is checked for (even or odd) specified by the O/$\overline{E}$ bit.

Bit 4—Parity Mode (O/$\overline{E}$): Selects either even or odd parity for use in parity addition checking.

The O/$\overline{E}$ bit setting is only valid when the PE bit is set to 1, enabling parity bit additio checking, in asynchronous mode. The O/$\overline{E}$ bit setting is invalid in synchronous mode, bit addition and checking is disabled in asynchronous mode, and when a multiprocess used.

Bit 4

O/$\overline{E}$	Description
0	Even parity*1
1	Odd parity*2

Notes: 1. When even parity is set, parity bit addition is performed in transmission so t number of 1 bits in the transmit character plus the parity bit is even.
In reception, a check is performed to see if the total number of 1 bits in the character plus the parity bit is even.
2. When odd parity is set, parity bit addition is performed in transmission so th number of 1 bits in the transmit character plus the parity bit is odd.
In reception, a check is performed to see if the total number of 1 bits in the character plus the parity bit is odd.

Notes: 1. In transmission, a single 1 bit (stop bit) is added to the end of a transmit cha
before it is sent.
2. In transmission, two 1 bits (stop bits) are added to the end of a transmit char
before it is sent.

In reception, only the first stop bit is checked, regardless of the STOP bit setting. If the
stop bit is 1, it is treated as a stop bit; if it is 0, it is treated as the start bit of the next tra
character.

Bit 2—Multiprocessor Mode (MP): Selects multiprocessor format. When multiproces
is selected, the PE bit and O/$\overline{E}$ bit parity settings are invalid. The MP bit setting is only
asynchronous mode; it is invalid in synchronous mode.

For details of the multiprocessor communication function, see section 15.3.3, Multipro
Communication Function.

Bit 2

MP	Description	
0	Multiprocessor function disabled	(
1	Multiprocessor format selected	

Bits 1 and 0—Clock Select 1 and 0 (CKS1, CKS0): These bits select the clock source
baud rate generator. The clock source can be selected from ϕ, $\phi/4$, $\phi/16$, and $\phi/64$, accor
setting of bits CKS1 and CKS0.

For the relation between the clock source, the bit rate register setting, and the baud rate
section 15.2.8, Bit Rate Register (BRR).

15.2.6 Serial Control Register (SCR)

Bit	7	6	5	4	3	2	1
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SCR is a register that performs enabling or disabling of SCI transfer operations, serial in asynchronous mode, and interrupt requests, and selection of the serial clock source.

SCR can be read or written to by the CPU at all times.

SCR is initialized to H'00 by a reset, and in standby mode, watch mode, subactive mo mode, and module stop mode.

Bit 7—Transmit Interrupt Enable (TIE): Enables or disables transmit-data-empty i (TXI) request generation when serial transmit data is transferred from TDR to TSR an flag in SSR is set to 1.

Bit 7

TIE	Description
0	Transmit-data-empty interrupt (TXI) request disabled*
1	Transmit-data-empty interrupt (TXI) request enabled

Note: * TXI interrupt request cancellation can be performed by reading 1 from the T then clearing it to 0, or clearing the TIE bit to 0.

1	Receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) re enabled

Note: * RXI and ERI interrupt request cancellation can be performed by reading 1 fr RDRF, FER, PER, or ORER flag, then clearing the flag to 0, or clearing the

Bit 5—Transmit Enable (TE): Enables or disables the start of serial transmission by t

Bit 5

TE	Description
0	Transmission disabled*1 (
1	Transmission enabled*2

Notes: 1. The TDRE flag in SSR is fixed at 1.

2. In this state, serial transmission is started when transmit data is written to TD TDRE flag in SSR is cleared to 0.

SMR setting must be performed to decide the transmission format before se bit to 1.

Bit 4—Receive Enable (RE): Enables or disables the start of serial reception by the S

Bit 4

RE	Description
0	Reception disabled*1 (
1	Reception enabled*2

Notes: 1. Clearing the RE bit to 0 does not affect the RDRF, FER, PER, and ORER fla retain their states.

2. Serial reception is started in this state when a start bit is detected in asynchr mode or serial clock input is detected in synchronous mode.

SMR setting must be performed to decide the reception format before setting to 1.

0	Multiprocessor interrupts disabled (normal reception performed) [Clearing conditions] • When the MPIE bit is cleared to 0 • When data with MPB = 1 is received
1	Multiprocessor interrupts enabled* Receive interrupt (RXI) requests, receive-error interrupt (ERI) requests, ar of the RDRF, FER, and ORER flags in SSR are disabled until data with th multiprocessor bit set to 1 is received.

Note: * When receive data including MPB = 0 is received, receive data transfer fro RDR, receive error detection, and setting of the RDRF, FER, and ORER fla is not performed. When receive data with MPB = 1 is received, the MPB bit set to 1, the MPIE bit is cleared to 0 automatically, and generation of RXI a interrupts (when the TIE and RIE bits in SCR are set to 1) and FER and OR setting is enabled.

Bit 2—Transmit End Interrupt Enable (TEIE): Enables or disables transmit-end in (TEI) request generation if there is no valid transmit data in TDR when the MSB is tra

Bit 2

TEIE	Description
0	Transmit-end interrupt (TEI) request disabled*
1	Transmit-end interrupt (TEI) request enabled*

Note: * TEI cancellation can be performed by reading 1 from the TDRE flag in SSR clearing it to 0 and clearing the TEND flag to 0, or clearing the TEIE bit to 0

Bits 1 and 0—Clock Enable 1 and 0 (CKE1, CKE0): These bits are used to select th source and enable or disable clock output from the SCK pin. The combination of the C CKE0 bits determines whether the SCK pin functions as an I/O port, the serial clock o the serial clock input pin.

The setting of the CKE0 bit, however, is only valid for internal clock operation (CKE asynchronous mode. The CKE0 bit setting is invalid in synchronous mode, and in the

		Synchronous mode	Internal clock/SCK pin functions as seria output*1
	1	Asynchronous mode	Internal clock/SCK pin functions as cloc
		Synchronous mode	Internal clock/SCK pin functions as seria output
1	0	Asynchronous mode	External clock/SCK pin functions as cloc
		Synchronous mode	External clock/SCK pin functions as seri input
	1	Asynchronous mode	External clock/SCK pin functions as cloc
		Synchronous mode	External clock/SCK pin functions as seri input

Notes: 1. Initial value
2. Outputs a clock of the same frequency as the bit rate.
3. Inputs a clock with a frequency 16 times the bit rate.

15.2.7 Serial Status Register (SSR)

Bit	7	6	5	4	3	2	1	
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	
Initial value	1	0	0	0	0	1	0	
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	

Note: * Only 0 can be written, to clear the flag.

SSR is an 8-bit register containing status flags that indicate the operating status of the S multiprocessor bits.

SSR can be read or written to by the CPU at all times. However, 1 cannot be written to TDRE, RDRF, ORER, PER, and FER. Also note that in order to clear these flags they read as 1 beforehand. The TEND flag and MPB flag are read-only flags and cannot be

	• When 0 is written in TDRE after reading TDRE = 1
	• When the DTC is activated by a TXI interrupt and writes data to TDR
1	[Setting conditions]
	• When the TE bit in SCR is 0
	• When data is transferred from TDR to TSR and data can be written to

Bit 6—Receive Data Register Full (RDRF): Indicates that the received data is stored

Bit 6

RDRF	Description
0	[Clearing conditions]
	• When 0 is written in RDRF after reading RDRF = 1
	• When the DTC is activated by an RXI interrupt and reads data from RD
1	[Setting condition]
	When serial reception ends normally and receive data is transferred from

Note: RDR and the RDRF flag are not affected and retain their previous values when detected during reception or when the RE bit in SCR is cleared to 0.
If reception of the next data is completed while the RDRF flag is still set to 1, an error will occur and the receive data will be lost.

	When the next serial reception is completed while RDRF = 1*2

Notes: 1. The ORER flag is not affected and retains its previous state when the RE bit
cleared to 0.
2. The receive data prior to the overrun error is retained in RDR, and the data r
subsequently is lost. Also, subsequent serial reception cannot be continued
ORER flag is set to 1. In synchronous mode, serial transmission cannot be c
either.

Bit 4—Framing Error (FER): Indicates that a framing error occurred during receptio
asynchronous mode, causing abnormal termination.

Bit 4

FER	Description
0	[Clearing condition] (In
	When 0 is written in FER after reading FER = 1
1	[Setting condition]
	When the SCI checks the stop bit at the end of the receive data when recep and the stop bit is 0*2

Notes: 1. The FER flag is not affected and retains its previous state when the RE bit ir
cleared to 0.
2. In 2-stop-bit mode, only the first stop bit is checked for a value of 0; the secc
is not checked. If a framing error occurs, the receive data is transferred to RI
RDRF flag is not set. Also, subsequent serial reception cannot be continued
FER flag is set to 1. In synchronous mode, serial transmission cannot be cor
either.

	When, in reception, the number of 1 bits in the receive data plus the parity match the parity setting (even or odd) specified by the O/$\overline{E}$ bit in SMR*2

Notes: 1. The PER flag is not affected and retains its previous state when the RE bit cleared to 0.

2. If a parity error occurs, the receive data is transferred to RDR but the RDRF set. Also, subsequent serial reception cannot be continued while the PER f 1. In synchronous mode, serial transmission cannot be continued, either.

Bit 2—Transmit End (TEND): Indicates that there is no valid data in TDR when the the transmit character is sent, and transmission has been ended.

The TEND flag is read-only and cannot be modified.

Bit 2

TEND	Description
0	[Clearing conditions] • When 0 is written in TDRE after reading TDRE = 1 • When the DTC is activated by a TXI interrupt and writes data to TDR
1	[Setting conditions] • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the last bit of a 1-byte serial transm

	When data with a 0 multiprocessor bit is received
1	[Setting condition] When data with a 1 multiprocessor bit is received

Note: * Retains its previous state when the RE bit in SCR is cleared to 0 with multipr format.

Bit 0—Multiprocessor Bit Transfer (MPBT): When transmission is performed using multiprocessor format in asynchronous mode, MPBT stores the multiprocessor bit to be the transmit data.

The MPBT bit setting is invalid when a multiprocessor format is not used, when not tra and in synchronous mode.

Bit 0

MPBT	Description
0	Data with a 0 multiprocessor bit is transmitted
1	Data with a 1 multiprocessor bit is transmitted

generator operating clock selected by bits CKS1 and CKS0 in SMR.

BRR can be read or written to by the CPU at all times.

BRR is initialized to H'FF by a reset, and in standby mode, watch mode, subactive mo
mode, and module stop mode.

As baud rate generator control is performed independently for each channel, different
be set for each channel.

Table 15.3 shows sample BRR settings in asynchronous mode, and table 15.4 shows s
settings in synchronous mode.

300	0	207	0.16	0	217	0.21	0	255	0.00	1	77
600	0	103	0.16	0	108	0.21	0	127	0.00	0	15
1200	0	51	0.16	0	54	–0.70	0	63	0.00	0	77
2400	0	25	0.16	0	26	1.14	0	31	0.00	0	38
4800	0	12	0.16	0	13	–2.48	0	15	0.00	0	19
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	—	—

	Operating Frequency ϕ (MHz)										
	ϕ = 3.6864 MHz			**ϕ = 4 MHz**			**ϕ = 4.9152 MHz**			**ϕ =**	
Bit Rate (bits/s)	**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	88
150	1	191	0.00	1	207	0.16	1	255	0.00	2	64
300	1	95	0.00	1	103	0.16	1	127	0.00	1	12
600	0	191	0.00	0	207	0.16	0	255	0.00	1	64
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	12
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	64
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	32
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	15
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

600	1	77	0.16	1	79	0.00	1	95	0.00	1	1
1200	0	155	0.16	0	159	0.00	0	191	0.00	0	2
2400	0	77	0.16	0	79	0.00	0	95	0.00	0	1
4800	0	38	0.16	0	39	0.00	0	47	0.00	0	5
9600	0	19	–2.34	0	19	0.00	0	23	0.00	0	2
19200	0	9	–2.34	0	9	0.00	0	11	0.00	0	1
31250	0	5	0.00	0	5	2.40	—	—	—	0	7
38400	0	4	–2.34	0	4	0.00	0	5	0.00	—	–

	Operating Frequency ϕ (MHz)										
	ϕ = 9.8304 MHz			ϕ = 10 MHz			ϕ = 12 MHz			ϕ = 1	
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N
110	2	174	–0.26	2	177	–0.25	2	212	0.03	2	2
150	2	127	0.00	2	129	0.16	2	155	0.16	2	1
300	1	255	0.00	2	64	0.16	2	77	0.16	2	7
600	1	127	0.00	1	129	0.16	1	155	0.16	1	1
1200	0	255	0.00	1	64	0.16	1	77	0.16	1	7
2400	0	127	0.00	0	129	0.16	0	155	0.16	0	1
4800	0	63	0.00	0	64	0.16	0	77	0.16	0	7
9600	0	31	0.00	0	32	–1.36	0	38	0.16	0	3
19200	0	15	0.00	0	15	1.73	0	19	–2.34	0	1
31250	0	9	–1.70	0	9	0.00	0	11	0.00	0	1
38400	0	7	0.00	0	7	1.73	0	9	–2.34	0	9

600	1	181	0.16	1	191	0.00	1	207	0.16	1	22
1200	1	90	0.16	1	95	0.00	1	103	0.16	1	11
2400	0	181	0.16	0	191	0.00	0	207	0.16	0	22
4800	0	90	0.16	0	95	0.00	0	103	0.16	0	11
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

	Operating Frequency ϕ (MHz)								
	ϕ = 18 MHz			**ϕ = 19.6608 MHz**			**ϕ = 20 MHz**		
Bit Rate (bits/s)	**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	116	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	116	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

1 k	1	124	1	249	2	124	—	—	2	249	–
2.5 k	0	199	1	99	1	199	1	249	2	99	2
5 k	0	99	0	199	1	99	1	124	1	199	1
10 k	0	49	0	99	0	199	0	249	1	99	1
25 k	0	19	0	39	0	79	0	99	0	159	0
50 k	0	9	0	19	0	39	0	49	0	79	0
100 k	0	4	0	9	0	19	0	24	0	39	0
250 k	0	1	0	3	0	7	0	9	0	15	0
500 k	0	0*	0	1	0	3	0	4	0	7	0
1 M			0	0*	0	1			0	3	0
2.5 M							0	0*			0
5 M											0

Legend:

Blank: Cannot be set.

—: Can be set, but there will be a degree of error.

*: Continuous transfer is not possible.

Note: As far as possible, the setting should be made so that the error is no more than

$$N = \frac{\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

Where B: Bit rate (bits/s)
N: BRR setting for baud rate generator ($0 \leq N \leq 255$)
ϕ: Operating frequency (MHz)
n: Baud rate generator input clock (n = 0 to 3)
(See the table below for the relation between n and the clock.)

		SMR Setting	
n	**Clock**	**CKS1**	**CKS0**
0	ϕ	0	0
1	$\phi/4$	0	1
2	$\phi/16$	1	0
3	$\phi/64$	1	1

The bit rate error in asynchronous mode is found from the following equation:

$$\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

2.4576	76800	0	0
3	93750	0	0
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
9.8304	307200	0	0
10	312500	0	0
12	375000	0	0
12.288	384000	0	0
14	437500	0	0
14.7456	460800	0	0
16	500000	0	0
17.2032	537600	0	0
18	562500	0	0
19.6608	614400	0	0
20	625000	0	0

4	1.0000	62500
4.9152	1.2288	76800
5	1.2500	78125
6	1.5000	93750
6.144	1.5360	96000
7.3728	1.8432	115200
8	2.0000	125000
9.8304	2.4576	153600
10	2.5000	156250
12	3.0000	187500
12.288	3.0720	192000
14	3.5000	218750
14.7456	3.6864	230400
16	4.0000	250000
17.2032	4.3008	268800
18	4.5000	281250
19.6608	4.9152	307200
20	5.0000	312500

12	2.0000	2000000.0
14	2.3333	2333333.3
16	2.6667	2666666.7
18	3.0000	3000000.0
20	3.3333	3333333.3

15.2.9 Serial Interface Mode Register (SCMR)

Bit	7	6	5	4	3	2	1
	—	—	—	—	SDIR	SINV	—
Initial value	1	1	1	1	0	0	1
Read/Write	—	—	—	—	R/W	R/W	—

SCMR is an 8-bit readable/writable register used to select SCI functions.

SCMR is initialized to H'F2 by a reset, and in standby mode, watch mode, subactive m subsleep mode, and module stop mode.

Bits 7 to 4—Reserved: These bits cannot be modified and are always read as 1.

Bit 3—Data Transfer Direction (SDIR): Selects the serial/parallel conversion forma

Bit 3

SDIR	Description
0	TDR contents are transmitted LSB-first
	Receive data is stored in RDR LSB-first
1	TDR contents are transmitted MSB-first
	Receive data is stored in RDR MSB-first

1	TDR contents are inverted before being transmitted Receive data is stored in RDR in inverted form

Bit 1—Reserved: This bit cannot be modified and is always read as 1.

Bit 0—Serial Communication Interface Mode Select (SMIF): Reserved bit. 1 should written in this bit.

Bit 0

SMIF	Description
0	Normal SCI mode
1	Reserved mode

15.2.10 Module Stop Control Register (MSTPCR)

	MSTPCRH								MSTPCRL					
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2
	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	MSTP9	MSTP8	MSTP7	MSTP6	MSTP5	MSTP4	MSTP3	MSTP2
Initial value	0	0	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mod When bit MSTP7, MSTP6, or MSTP5 is set to 1, SCI0, SCI1, or SCI2 operation, respe stops at the end of the bus cycle and a transition is made to module stop mode. For deta section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

Bit 6—Module Stop (MSTP6): Specifies the SCI1 module stop mode.

MSTPCRL

Bit 6	
MSTP6	**Description**
0	SCI1 module stop mode is cleared
1	SCI1 module stop mode is set

Bit 5—Module Stop (MSTP5): Specifies the SCI2 module stop mode.

MSTPCRL

Bit 5	
MSTP5	**Description**
0	SCI2 module stop mode is cleared
1	SCI2 module stop mode is set

converter.

KBCOMP is initialized to H'00 by a reset and in hardware standby mode.

Bit 7—IrDA Enable (IrE): Specifies normal SCI operation or IrDA operation for SCI input/output.

Bit 7	
IrE	**Description**
0	The TxD2/IrTxD and RxD2/IrRxD pins function as TxD2 and RxD2
1	The TxD2/IrTxD and RxD2/IrRxD pins function as IrTxD and IrRxD

Bits 6 to 4—IrDA Clock Select 2 to 0 (IrCKS2 to IrCKS0): These bits specify the hi width in IrTxD output pulse encoding when the IrDA function is enabled.

Bit 6	Bit 5	Bit 4	
IrCKS2	**IrCKS1**	**IrCKS0**	**Description**
0	0	0	B × 3/16 (3/16 of the bit rate)
		1	φ/2
	1	0	φ/4
		1	φ/8
1	0	0	φ/16
		1	φ/32
	1	0	φ/64
		1	φ/128

Bits 3 to 0—Keyboard Comparator Control: See the description in section 20, A/D

Selection of asynchronous or synchronous mode and the transmission format is made as shown in table 15.8. The SCI clock is determined by a combination of the C/$\overline{A}$ bit i the CKE1 and CKE0 bits in SCR, as shown in table 15.9.

Asynchronous Mode:

- Data length: Choice of 7 or 8 bits
- Choice of parity addition, multiprocessor bit addition, and addition of 1 or 2 stop b combination of these parameters determines the transfer format and character leng
- Detection of framing, parity, and overrun errors, and breaks, during reception
- Choice of internal or external clock as SCI clock source
  - When internal clock is selected:

    The SCI operates on the baud rate generator clock and a clock with the same fr the bit rate can be output
  - When external clock is selected:

    A clock with a frequency of 16 times the bit rate must be input (the built-in bau generator is not used)

Synchronous Mode:

- Transfer format: Fixed 8-bit data
- Detection of overrun errors during reception
- Choice of internal or external clock as SCI clock source
  - When internal clock is selected:

    The SCI operates on the baud rate generator clock and a serial clock is output c
  - When external clock is selected:

    The built-in baud rate generator is not used, and the SCI operates on the input s

			1	0				Yes
				1				
	1		0	0		7-bit data		No
				1				
			1	0				Yes
				1				
	0	1	—	0	Asynchronous mode (multi-processor format)	8-bit data	Yes	No
			—	1				
	1		—	0		7-bit data		
			—	1				
1	—	—	—	—	Synchronous mode	8-bit data	No	

Table 15.9 SMR and SCR Settings and SCI Clock Source Selection

SMR	SCR Setting			SCI Transfer Clock	
Bit 7	Bit 1	Bit 0			
C/$\overline{A}$	CKE1	CKE0	Mode	Clock Source	SCK Pin Function
0	0	0	Asynchronous mode	Internal	SCI does not use SCK pin
		1			Outputs clock with same freque rate
	1	0		External	Inputs clock with frequency of 1 the bit rate
		1			
1	0	0	Synchronous mode	Internal	Outputs serial clock
		1			
	1	0		External	Inputs serial clock
		1			

that data can be read or written during transmission or reception, enabling continuous transfer.

Figure 15.2 shows the general format for asynchronous serial communication.

In asynchronous serial communication, the transmission line is usually held in the mar level). The SCI monitors the transmission line, and when it goes to the space state (low recognizes a start bit and starts serial communication.

One serial communication character consists of a start bit (low level), followed by dat first order), a parity bit (high or low level), and finally one or two stop bits (high level

In asynchronous mode, the SCI performs synchronization at the falling edge of the sta reception. The SCI samples the data on the 8th pulse of a clock with a frequency of 16 length of one bit, so that the transfer data is latched at the center of each bit.

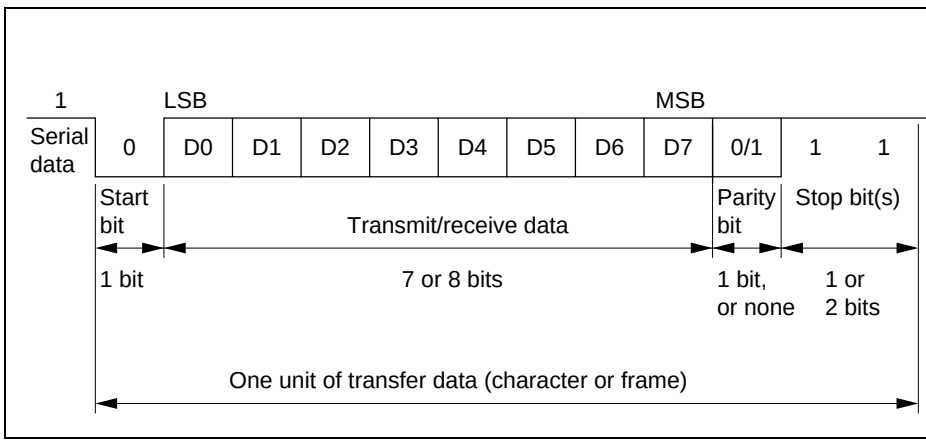


Figure 15.2 Data Format in Asynchronous Communication (Example with 8-Bit Data, Parity, Two Stop Bits)

0	0	0	0	S	8-bit data			STOP
0	0	0	1	S	8-bit data			STOP
0	1	0	0	S	8-bit data			P
0	1	0	1	S	8-bit data			P
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
0	—	1	0	S	8-bit data			MPB
0	—	1	1	S	8-bit data			MPB
1	—	1	0	S	7-bit data		MPB	STOP
1	—	1	1	S	7-bit data		MPB	STOP

Legend:
S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit

When the SCI is operated on an internal clock, the clock can be output from the SCK
frequency of the clock output in this case is equal to the bit rate, and the phase is such
rising edge of the clock is at the center of each transmit data bit, as shown in figure 15

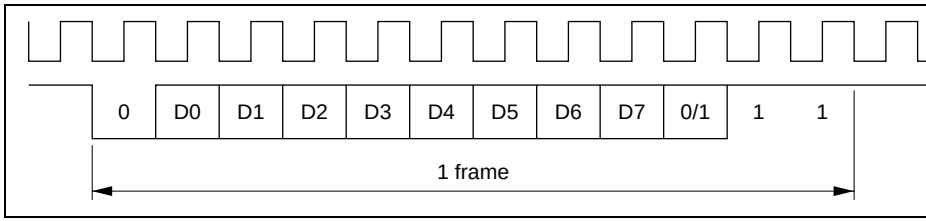


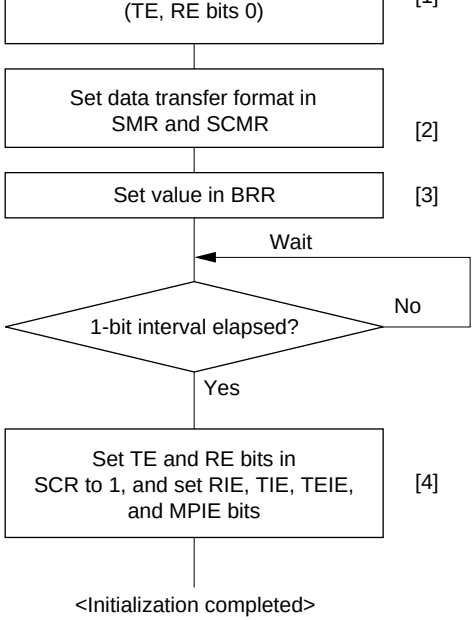
**Figure 15.3 Relation between Output Clock and Transfer Data Phas
(Asynchronous Mode)**

Data Transfer Operations

SCI Initialization (Asynchronous Mode): Before transmitting and receiving data, fi
TE and RE bits in SCR to 0, then initialize the SCI as described below.

When the operating mode, transfer format, etc., is changed, the TE and RE bits must b
0 before making the change using the following procedure. When the TE bit is cleared
TDRE flag is set to 1 and TSR is initialized. Note that clearing the RE bit to 0 does no
contents of the RDRF, PER, FER, and ORER flags, or the contents of RDR.

When an external clock is used the clock should not be stopped during operation, incl
initialization, since operation is uncertain.



immediately after SCR setting made.

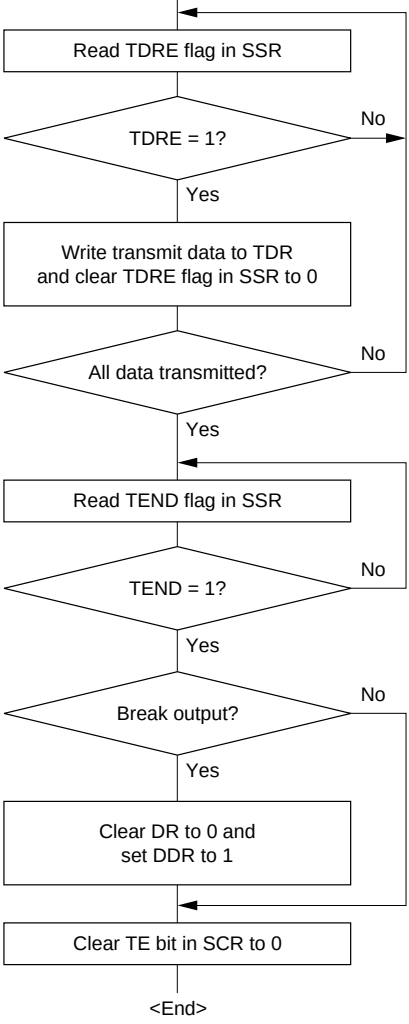
[2] Set the data transfer format in and SCMR.

[3] Write a value corresponding to bit rate to BRR. This is not necessary if an external clock used.

[4] Wait at least one bit interval, th set the TE bit or RE bit in SCR Also set the RIE, TIE, TEIE, an MPIE bits.

Setting the TE and RE bits ena the TxD and RxD pins to be us

Figure 15.4 Sample SCI Initialization Flowchart



After the TE bit is set to 1, one frame of 1s is output and transmission is enabled.

[2] SCI status check and transmit d write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear t 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 then clear the TDRE flag to 0. Check and clearing of the TDRE flag is automatic when the DTC is activated by a transmit-data-em interrupt (TXI) request, and data written to TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set DDR for the po corresponding to the TxD pin to clear DR to 0, then clear the TE in SCR to 0.

Figure 15.5 Sample Serial Transmission Flowchart

The serial transmit data is sent from the TXD pin in the following order.

a. Start bit:
   One 0-bit is output.
b. Transmit data:
   8-bit or 7-bit data is output in LSB-first order.
c. Parity bit or multiprocessor bit:
   One parity bit (even or odd parity), or one multiprocessor bit is output.
   A format in which neither a parity bit nor a multiprocessor bit is output can also
   selected.
d. Stop bit(s):
   One or two 1-bits (stop bits) are output.
e. Mark state:
   1 is output continuously until the start bit that starts the next transmission is sent

3. The SCI checks the TDRE flag at the timing for sending the stop bit.
   If the TDRE flag is cleared to 0, the data is transferred from TDR to TSR, the stop b
   and then serial transmission of the next frame is started.
   If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, the stop bit is sent, an
   mark state is entered in which 1 is output continuously. If the TEIE bit in SCR is se
   time, a TEI interrupt request is generated.

Figure 15.6 shows an example of the operation for transmission in asynchronous mode.

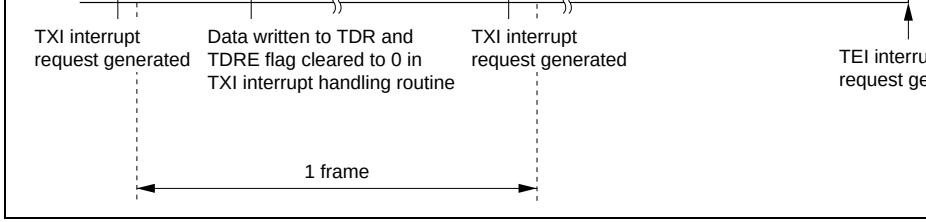
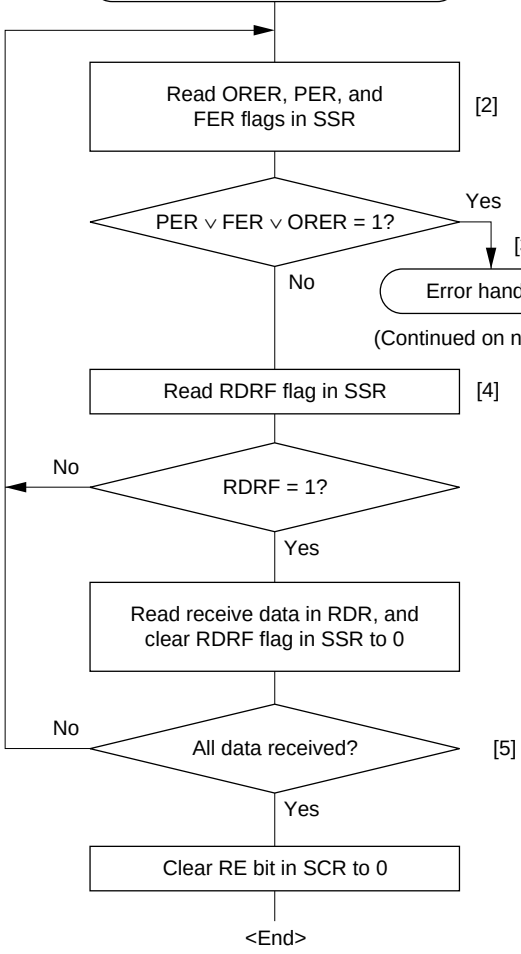


**Figure 15.6 Example of Operation in Transmission in Asynchronous M
(Example with 8-Bit Data, Parity, One Stop Bit)**



[2] [3] Receive error handling break detection:
If a receive error occurs, ORER, PER, and FER fla SSR to identify the error. performing the appropriat handling, ensure that the PER, and FER flags are a cleared to 0. Reception ca be resumed if any of thes are set to 1. In the case o framing error, a break can detected by reading the v the input port correspondi the RxD pin.

[4] SCI status check and rec data read :
Read SSR and check tha = 1, then read the receive RDR and clear the RDRF 0. Transition of the RDRF from 0 to 1 can also be id by an RXI interrupt.

[5] Serial reception continuat procedure:
To continue serial recepti before the stop bit for the frame is received, read th RDRF flag, read RDR, an the RDRF flag to 0. The R flag is cleared automatica when the DTC is activate RXI interrupt and the RDR is read.

Figure 15.7 Sample Serial Reception Data Flowchart

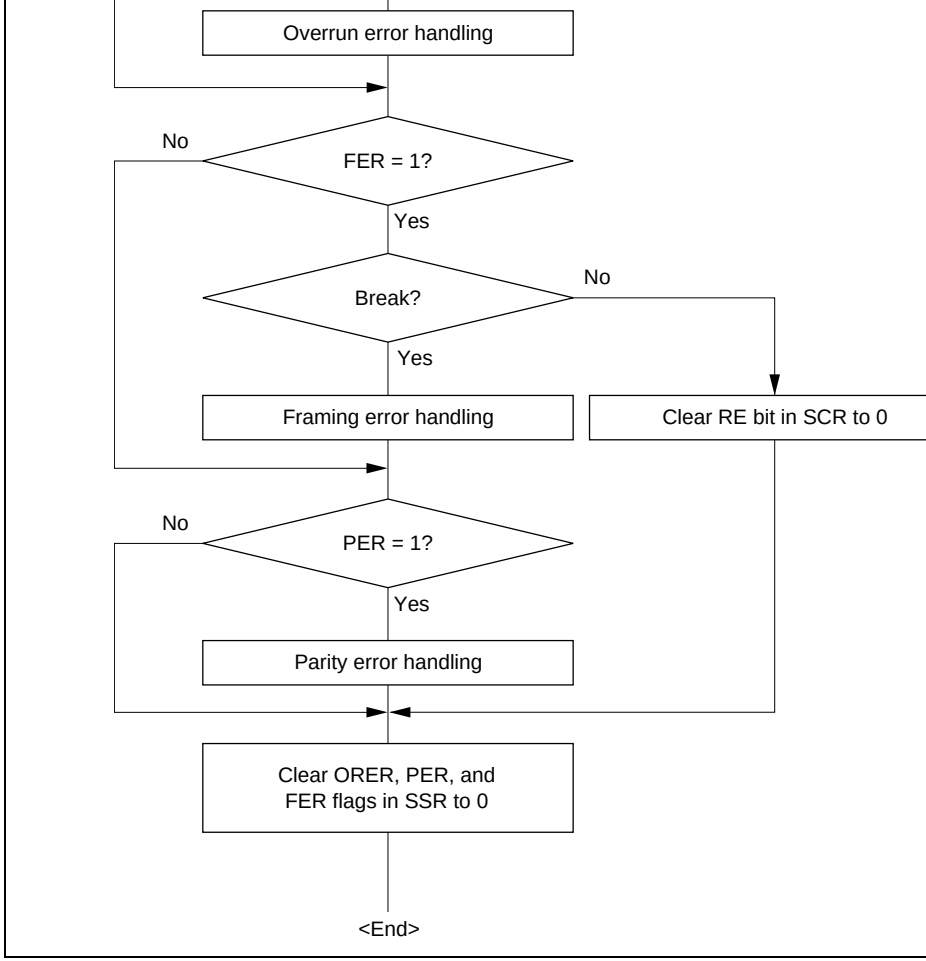


Figure 15.7 Sample Serial Reception Data Flowchart (cont)

a. Parity check:

   The SCI checks whether the number of 1 bits in the receive data agrees with the (even or odd) set in the O/$\overline{E}$ bit in SMR.

b. Stop bit check:

   The SCI checks whether the stop bit is 1.

   If there are two stop bits, only the first is checked.

c. Status check:

   The SCI checks whether the RDRF flag is 0, indicating that the receive data can transferred from RSR to RDR.

If all the above checks are passed, the RDRF flag is set to 1, and the receive data is RDR.

If a receive error* is detected in the error check, the operation is as shown in table 1

Note: * Subsequent receive operations cannot be performed when a receive error ha Also note that the RDRF flag is not set to 1 in reception, and so the error fla cleared to 0.

4. If the RIE bit in SCR is set to 1 when the RDRF flag changes to 1, a receive-data-fu (RXI) request is generated.

   Also, if the RIE bit in SCR is set to 1 when the ORER, PER, or FER flag changes to receive-error interrupt (ERI) request is generated.

from the parity (even or odd) set in SMR | from RSR to R

Figure 15.8 shows an example of the operation for reception in asynchronous mode.

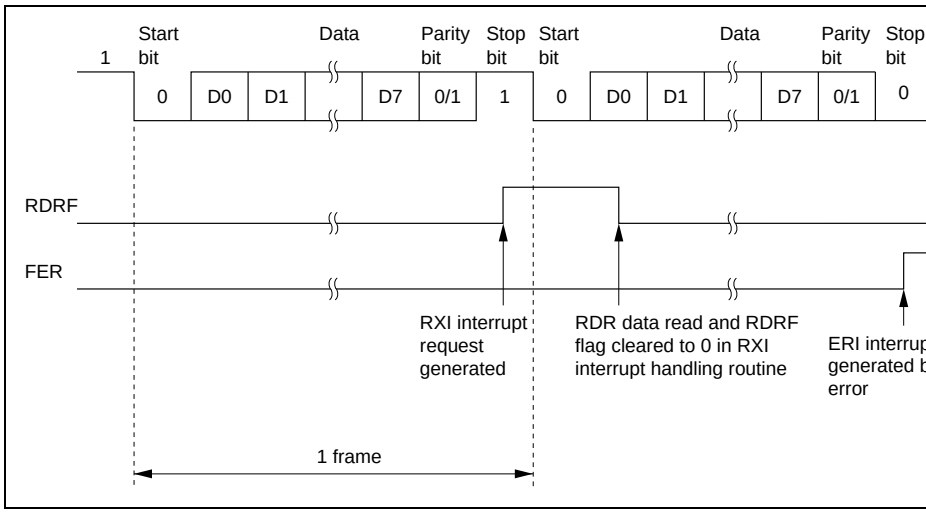


Figure 15.8 Example of SCI Operation in Reception (Example with 8-Bit Data, Parity, One Stop Bit)

The serial communication cycle consists of two component cycles: an ID transmission
which specifies the receiving station, and a data transmission cycle. The multiprocesso
to differentiate between the ID transmission cycle and the data transmission cycle.

The transmitting station first sends the ID of the receiving station with which it wants t
serial communication as data with a 1 multiprocessor bit added. It then sends transmit d
with a 0 multiprocessor bit added.

The receiving station skips the data until data with a 1 multiprocessor bit is sent.

When data with a 1 multiprocessor bit is received, the receiving station compares that d
own ID. The station whose ID matches then receives the data sent next. Stations whose
not match continue to skip the data until data with a 1 multiprocessor bit is again receiv
way, data communication is carried out among a number of processors.

Figure 15.9 shows an example of inter-processor communication using a multiprocesso

Data Transfer Format

There are four data transfer formats.

When a multiprocessor format is specified, the parity bit specification is invalid.

For details, see table 15.10.

Clock

See the section on asynchronous mode.

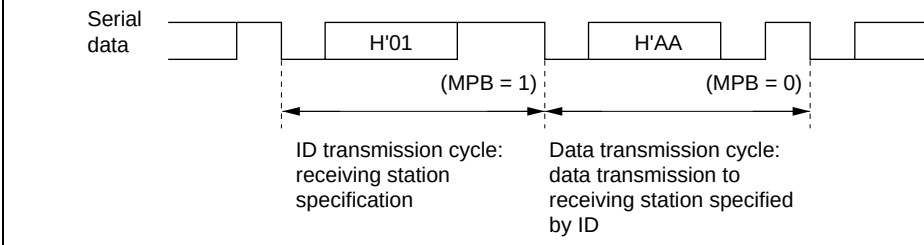
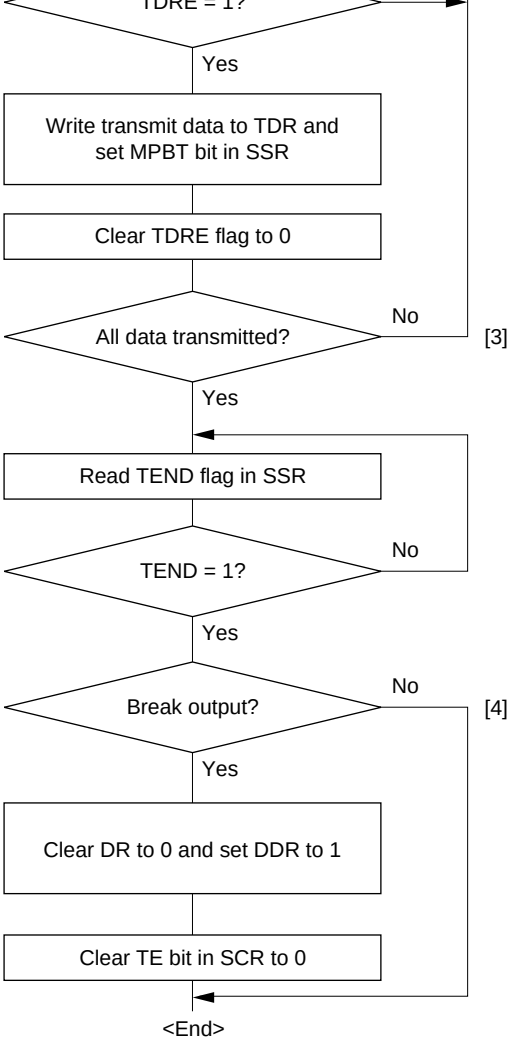


Figure 15.9 Example of Inter-Processor Communication Using Multiprocesso (Transmission of Data H'AA to Receiving Station A)

Data Transfer Operations

Multiprocessor Serial Data Transmission: Figure 15.10 shows a sample flowchart f multiprocessor serial data transmission.

The following procedure should be used for multiprocessor serial data transmission.



data write.
Read SSR and check that th
TDRE flag is set to 1, then w
transmit data to TDR. Set th
MPBT bit in SSR to 0 or 1.
Finally, clear the TDRE flag

[3] Serial transmission continua
procedure:
To continue serial transmiss
be sure to read 1 from the T
flag to confirm that writing is
possible, then write data to T
and then clear the TDRE flag
0. Checking and clearing of
TDRE flag is automatic when
DTC is activated by a transm
data-empty interrupt (TXI)
request, and data is written t
TDR.

[4] Break output at the end of se
transmission:
To output a break in serial
transmission, set the port DD
1, clear DR to 0, then clear t
TE bit in SCR to 0.

Figure 15.10 Sample Multiprocessor Serial Transmission Flowchart

The serial transmit data is sent from the TXD pin in the following order.

a. Start bit:
   One 0-bit is output.
b. Transmit data:
   8-bit or 7-bit data is output in LSB-first order.
c. Multiprocessor bit
   One multiprocessor bit (MPBT value) is output.
d. Stop bit(s):
   One or two 1-bits (stop bits) are output.
e. Mark state:
   1 is output continuously until the start bit that starts the next transmission is ser

3. The SCI checks the TDRE flag at the timing for sending the stop bit.
   If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, the stop bit
   then serial transmission of the next frame is started.
   If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, the stop bit is sent, a
   mark state is entered in which 1 is output continuously. If the TEIE bit in SCR is s
   time, a transmit-end interrupt (TEI) request is generated.

Figure 15.11 shows an example of SCI operation for transmission using a multiproces

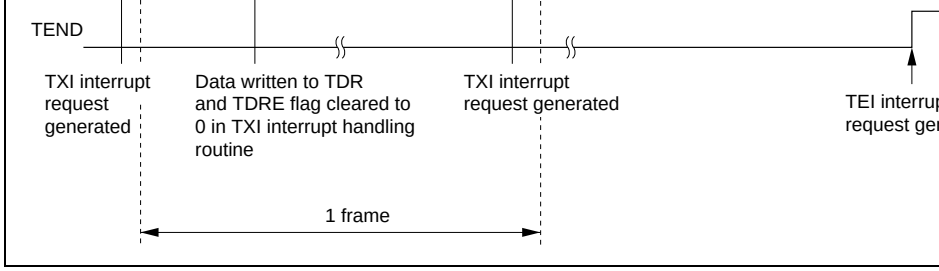
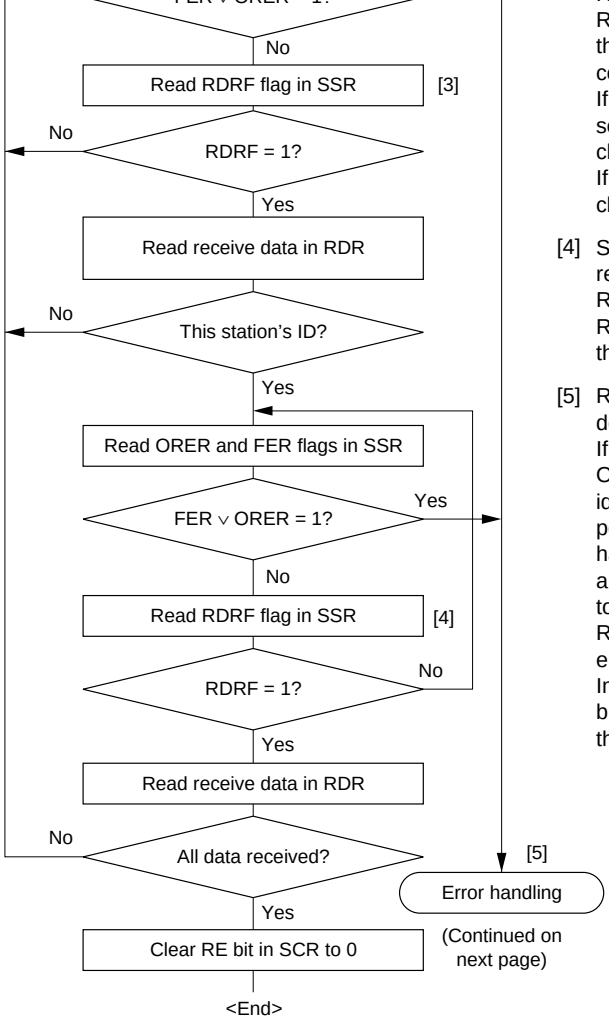


Figure 15.11 Example of SCI Operation in Transmission (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

Multiprocessor Serial Data Reception: Figure 15.12 shows a sample flowchart for multiprocessor serial reception.

The following procedure should be used for multiprocessor serial data reception.



RDRF flag is set to 1, then
the receive data in RDR an
compare it with this station'
If the data is not this statio
set the MPIE bit to 1 again,
clear the RDRF flag to 0.
If the data is this station's I
clear the RDRF flag to 0.

[4] SCI status check and data
reception:
Read SSR and check that
RDRF flag is set to 1, then
the data in RDR.

[5] Receive error handling and
detection:
If a receive error occurs, re
ORER and FER flags in SS
identify the error. After
performing the appropriate
handling, ensure that the C
and FER flags are both cle
to 0.
Reception cannot be resum
either of these flags is set t
In the case of a framing err
break can be detected by r
the RxD pin value.

Figure 15.12 Sample Multiprocessor Serial Reception Flowchart

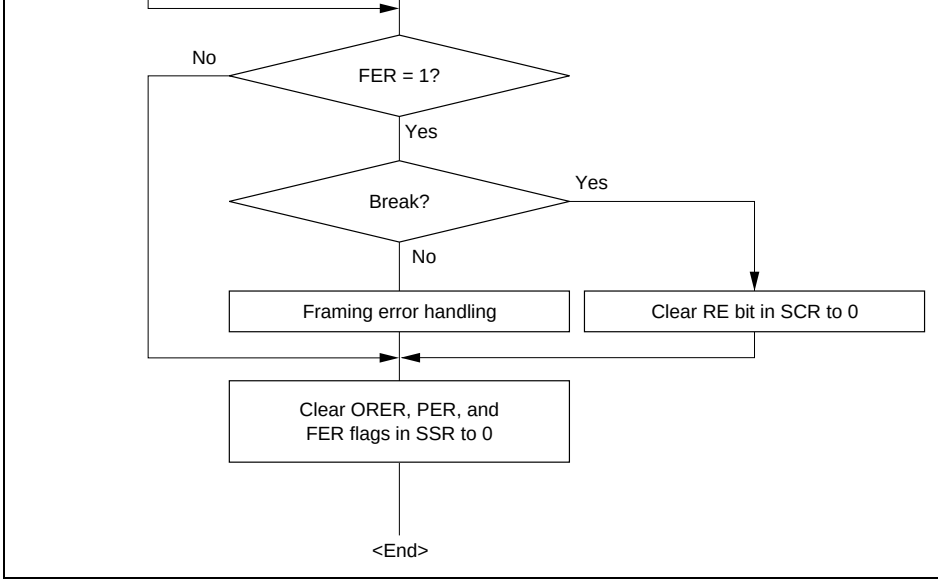


Figure 15.12 Sample Multiprocessor Serial Reception Flowchart (cont

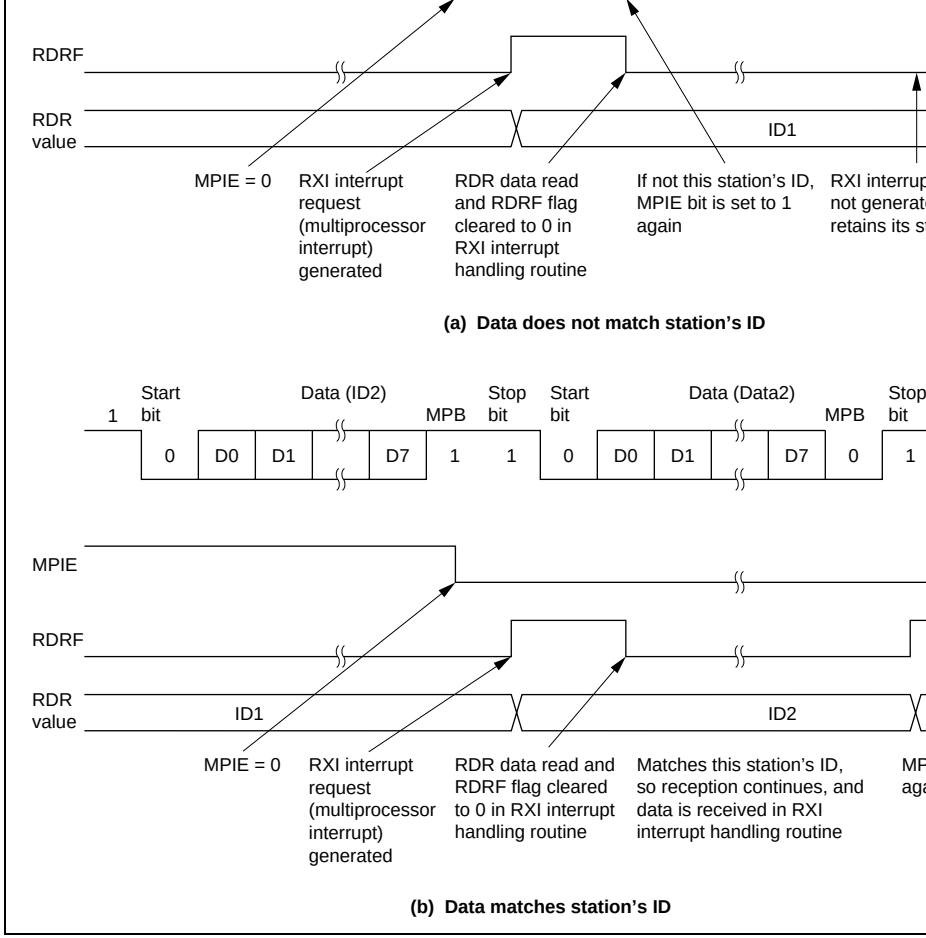


Figure 15.13 Example of SCI Operation in Reception (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

Figure 15.14 shows the general format for synchronous serial communication.

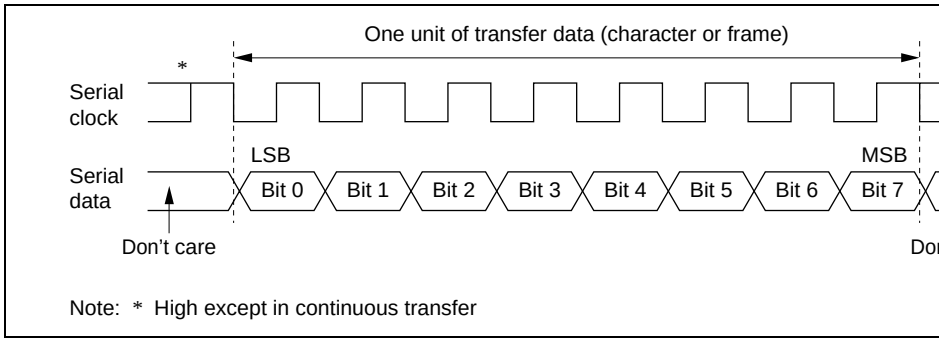


Figure 15.14 Data Format in Synchronous Communication

In synchronous serial communication, data on the transmission line is output from one
of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial cl

In synchronous serial communication, one character consists of data output starting wit
and ending with the MSB. After the MSB is output, the transmission line holds the MS

In synchronous mode, the SCI receives data in synchronization with the rising edge of
clock.

Data Transfer Format

A fixed 8-bit data format is used.

No parity or multiprocessor bits are added.

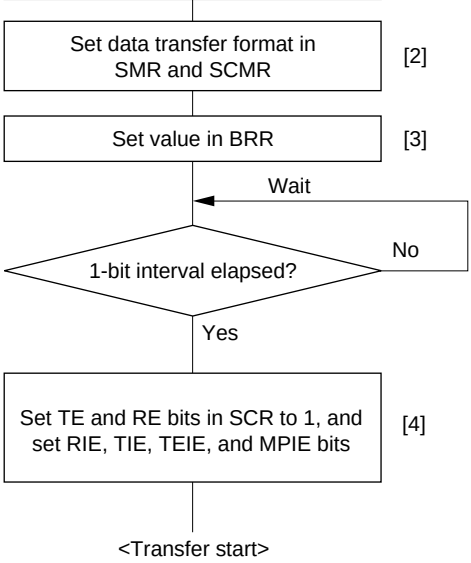
performed the clock is fixed high. When only receive operations are performed, howe
serial clock is output until an overrun error occurs or the RE bit is cleared to 0. To per
operations in units of one character, select an external clock as the clock source.

Data Transfer Operations

SCI Initialization (Synchronous Mode): Before transmitting and receiving data, firs
TE and RE bits in SCR to 0, then initialize the SCI as described below.

When the operating mode, transfer format, etc., is changed, the TE and RE bits must b
0 before making the change using the following procedure. When the TE bit is cleared
TDRE flag is set to 1 and TSR is initialized. Note that clearing the RE bit to 0 does no
settings of the RDRF, PER, FER, and ORER flags, or the contents of RDR.

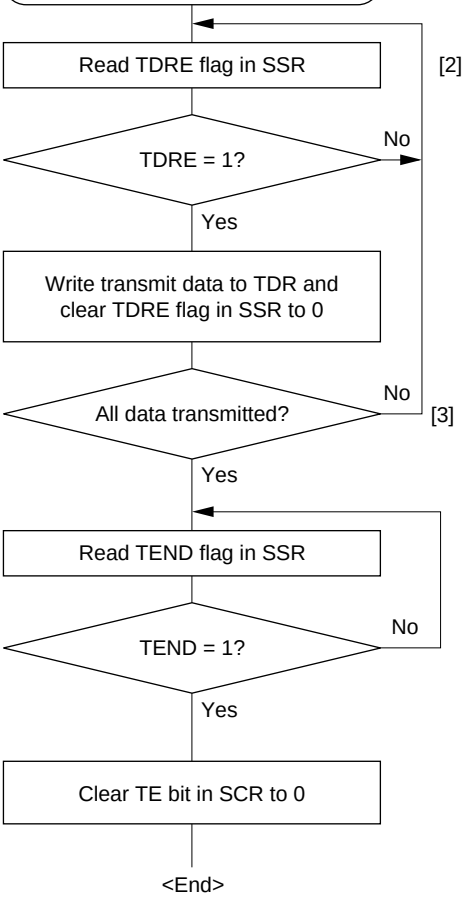
Figure 15.15 shows a sample SCI initialization flowchart.



[4] Wait at least one bit interval, then
the TE bit or RE bit in SCR to 1.
Also set the RIE, TIE, TEIE, and M
bits.
Setting the TE and RE bits enable
TxD and RxD pins to be used.

Note: In simultaneous transmit and receive operations, the TE bit and RE bit should both be
to 0 or set to 1 simultaneously.

Figure 15.15 Sample SCI Initialization Flowchart



pin.

[2] SCI status check and transmit
write:
Read SSR and check that the
flag is set to 1, then write trans
to TDR and clear the TDRE fla

[3] Serial transmission continuatio
procedure:
To continue serial transmissio
sure to read 1 from the TDRE
confirm that writing is possible
write data to TDR, and then cl
TDRE flag to 0.
Checking and clearing of the T
flag is automatic when the DT
activated by a transmit-data-e
interrupt (TXI) request and dat
written to TDR.

Figure 15.16 Sample Serial Transmission Flowchart

external clock has been specified, data is output synchronized with the input clock.

The serial transmit data is sent from the TxD pin starting with the LSB (bit 0) and e the MSB (bit 7).

3. The SCI checks the TDRE flag at the timing for sending the MSB (bit 7).

   If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, and serial tra of the next frame is started.

   If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, the MSB (bit 7) is ser TxD pin maintains its state.

   If the TEIE bit in SCR is set to 1 at this time, a transmit-end interrupt (TEI) request generated.

4. After completion of serial transmission, the SCK pin is held in a constant state.

Figure 15.17 shows an example of SCI operation in transmission.

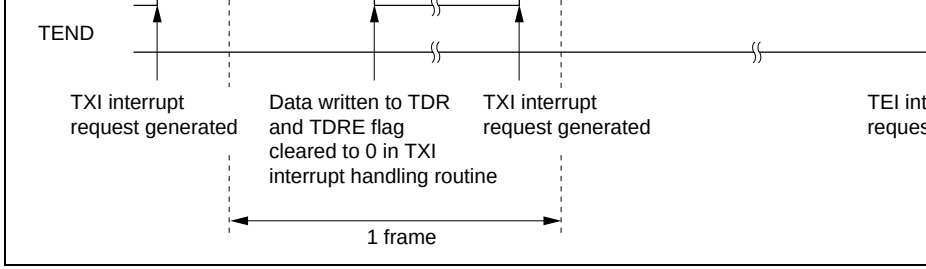


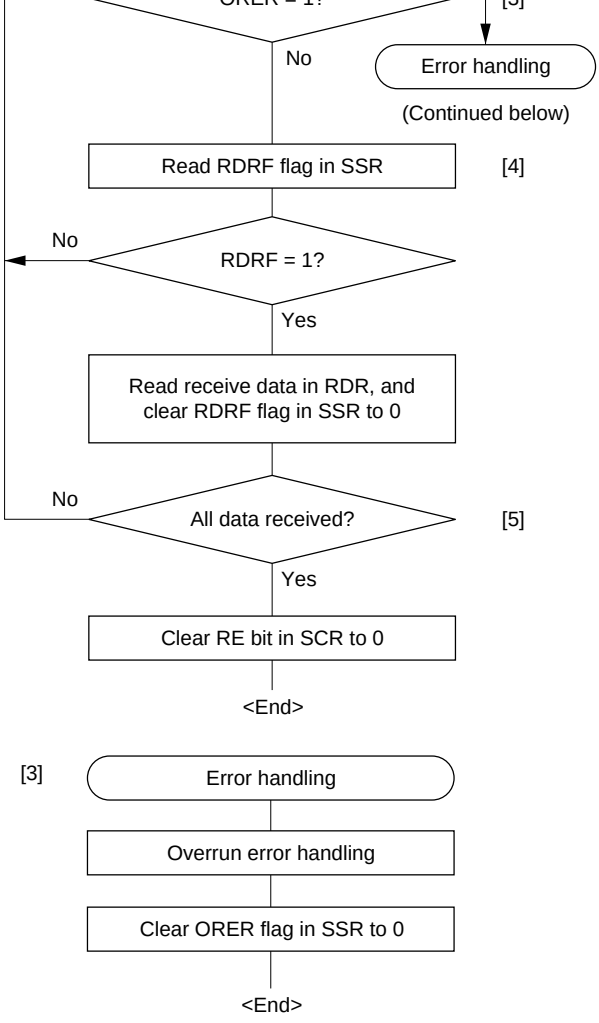
Figure 15.17 Example of SCI Operation in Transmission

Serial Data Reception (Synchronous Mode): Figure 15.18 shows a sample flowchar reception.

The following procedure should be used for serial data reception.

When changing the operating mode from asynchronous to synchronous, be sure to che ORER, PER, and FER flags are all cleared to 0.

The RDRF flag will not be set if the FER or PER flag is set to 1, and neither transmit operations will be possible.



[4] SCI status check and rec
data read:
Read SSR and check tha
RDRF flag is set to 1, the
the receive data in RDR a
clear the RDRF flag to 0.
Transition of the RDRF fl
0 to 1 can also be identifi
an RXI interrupt.

[5] Serial reception continua
procedure:
To continue serial recepti
before the MSB (bit 7) of
current frame is received
reading the RDRF flag, re
RDR, and clearing the RD
to 0. The RDRF flag is cle
automatically when the D
activated by a receive-da
interrupt (RXI) request an
RDR value is read.

Figure 15.18 Sample Serial Reception Flowchart

receive error is detected in the error check, the operation is as shown in table 15.11

Neither transmit nor receive operations can be performed subsequently when a rec has been found in the error check.

3. If the RIE bit in SCR is set to 1 when the RDRF flag changes to 1, a receive-data-f (RXI) request is generated.

   Also, if the RIE bit in SCR is set to 1 when the ORER flag changes to 1, a receive- interrupt (ERI) request is generated.

Figure 15.19 shows an example of SCI operation in reception.

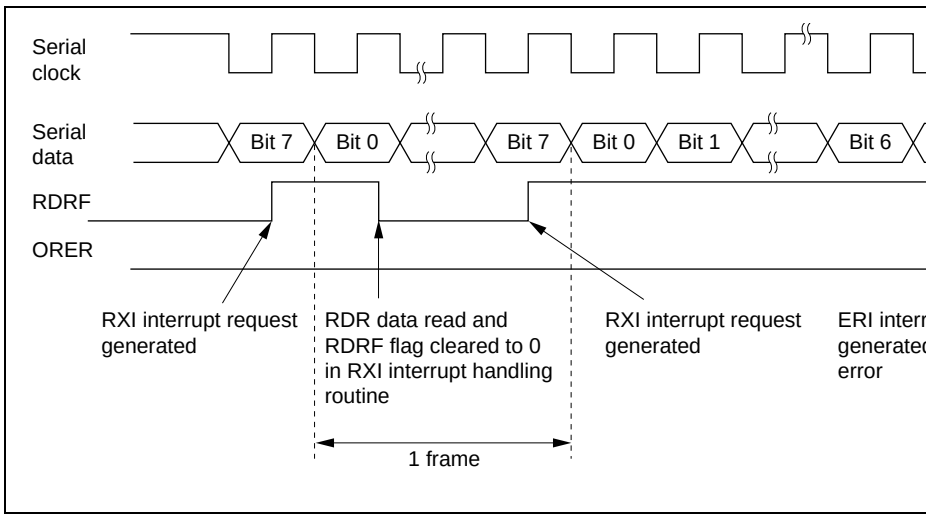
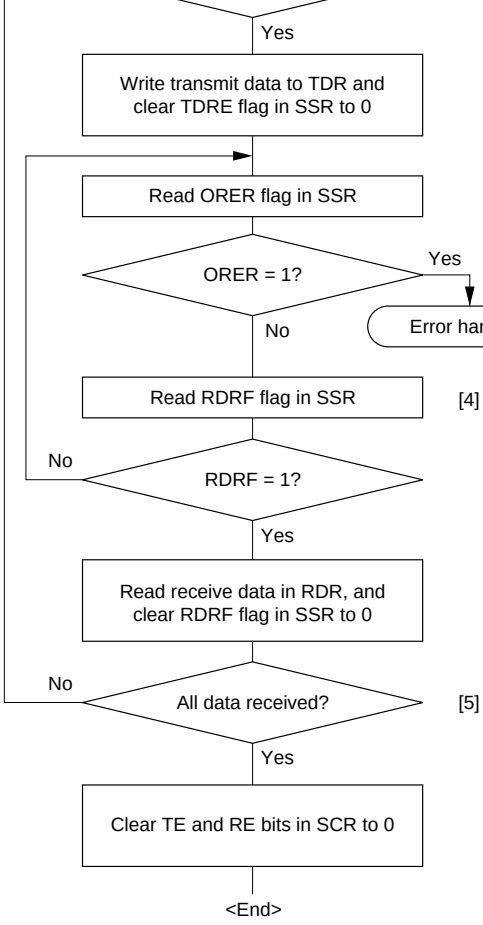


Figure 15.19 Example of SCI Operation in Reception

Simultaneous Serial Data Transmission and Reception (Synchronous Mode): Fig shows a sample flowchart for simultaneous serial transmit and receive operations.

The following procedure should be used for simultaneous serial data transmit and rece operations.



Note: When switching from transmit or receive operation to simultaneous transmit and receive operations, first clear the TE bit and RE bit to 0, then set both these bits to 1 simultaneously.

[3] Receive error handling:
If a receive error occurs, rea
ORER flag in SSR , and aft
performing the appropriate
handling, clear the ORER fl
Transmission/reception can
resumed if the ORER flag is
1.

[4] SCI status check and receiv
read:
Read SSR and check that t
RDRF flag is set to 1, then
receive data in RDR and cle
RDRF flag to 0. Transition o
RDRF flag from 0 to 1 can a
identified by an RXI interrup

[5] Serial transmission/receptio
continuation procedure:
To continue serial transmiss
reception, before the MSB (
the current frame is receive
reading the RDRF flag, read
RDR, and clearing the RDR
0. Also, before the MSB (bit
the current frame is transmi
read 1 from the TDRE flag t
confirm that writing is possi
write data to TDR and clear
TDRE flag to 0.
Checking and clearing of th
flag is automatic when the D
activated by a transmit-data
interrupt (TXI) request and
written to TDR. Also, the RD
is cleared automatically whe
DTC is activated by a receiv
full interrupt (RXI) request a
RDR value is read.

Figure 15.20 Sample Flowchart of Simultaneous Serial Transmit and Receive O

In the IrDA specification version 1.0 system, communication is started at a transfer ra
bps, and subsequently the transfer rate can be varied as necessary. As the IrDA interfa
LSI does not include a function for varying the transfer rate automatically, the transfe
must be changed by software.

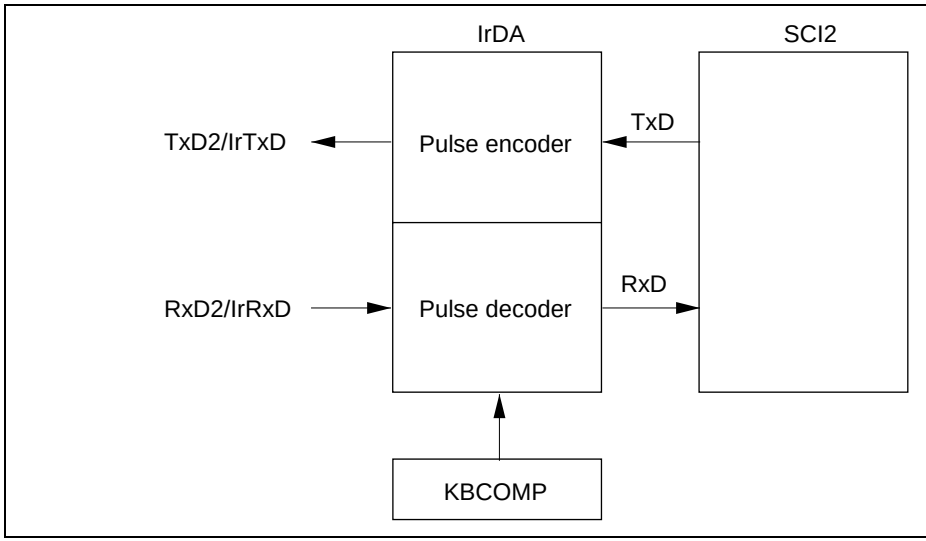


Figure 15.21 Block Diagram of IrDA Function

Transmission

In transmission, the output signal (UART frame) from the SCI is converted to an IR fr
IrDA interface (see figure 15.22).

When the serial data is 0, a high-level pulses of 3/16 the bit rate (interval equivalent to
of one bit) is output (initial value). The high-level pulse can be varied according to the
bits IrCKS2 to IrCKS0 in KBCOMP.

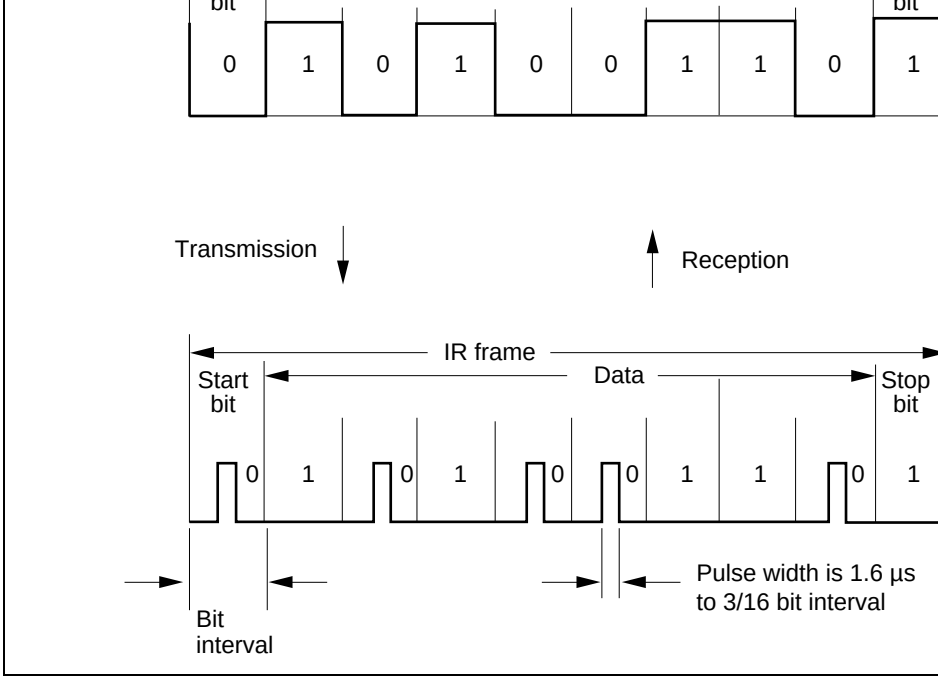


Figure 15.22 IrDA Transmit/Receive Operations

Reception

In reception, IR frame data is converted to a UART frame by the IrDA interface, and in SCI.

When a high-level pulse is detected, 0 data is output, and if there is no pulse during a o interval, 1 data is output. Note that a pulse shorter than the minimum pulse width of 1.4 be identified as a 0 signal.

Operating Frequency	2400	9600	19200	38400	57600
ϕ (MHz)	78.13	19.53	9.77	4.88	3.26
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
4.9152	011	011	011	011	011
5	011	011	011	011	011
6	100	100	100	100	100
6.144	100	100	100	100	100
7.3728	100	100	100	100	100
8	100	100	100	100	100
9.8304	100	100	100	100	100
10	100	100	100	100	100
12	101	101	101	101	101
12.288	101	101	101	101	101
14	101	101	101	101	101
14.7456	101	101	101	101	101
16	101	101	101	101	101
16.9344	101	101	101	101	101
17.2032	101	101	101	101	101
18	101	101	101	101	101
19.6608	101	101	101	101	101
20	101	101	101	101	101

Legend:

—: An SCI bit rate setting cannot be mode.

in SSR is set to 1, a TEI interrupt request is generated. A TXI interrupt can activate the
perform data transfer. The TDRE flag is cleared to 0 automatically when data transfer i
performed by the DTC. The DTC cannot be activated by a TEI interrupt request.

When the RDRF flag in SSR is set to 1, an RXI interrupt request is generated. When th
PER, or FER flag in SSR is set to 1, an ERI interrupt request is generated. An RXI inte
activate the DTC to perform data transfer. The RDRF flag is cleared to 0 automatically
transfer is performed by the DTC. The DTC cannot be activated by an ERI interrupt re

Table 15.13 SCI Interrupt Sources

Channel	Interrupt Source	Description	DTC Activation
0	ERI	Receive error (ORER, FER, or PER)	Not possible
	RXI	Receive data register full (RDRF)	Possible
	TXI	Transmit data register empty (TDRE)	Possible
	TEI	Transmit end (TEND)	Not possible
1	ERI	Receive error (ORER, PER, or PER)	Not possible
	RXI	Receive data register full (RDRF)	Possible
	TXI	Transmit data register empty (TDRE)	Possible
	TEI	Transmit end (TEND)	Not possible
2	ERI	Receive error (ORER, PER, or PER)	Not possible
	RXI	Receive data register full (RDRF)	Possible
	TXI	Transmit data register empty (TDRE)	Possible
	TEI	Transmit end (TEND)	Not possible

Note: * The table shows the initial state immediately after a reset. Relative channel p
can be changed by the interrupt controller.

The TEI interrupt is requested when the TEND flag is set to 1 while the TEIE bit is set
TEND flag is cleared at the same time as the TDRE flag. Consequently, if a TEI interru

Relation between Writes to TDR and the TDRE Flag

The TDRE flag in SSR is a status flag that indicates that transmit data has been transf
TDR to TSR. When the SCI transfers data from TDR to TSR, the TDRE flag is set to

Data can be written to TDR regardless of the state of the TDRE flag. However, if new
written to TDR when the TDRE flag is cleared to 0, the data stored in TDR will be los
not yet been transferred to TSR. It is therefore essential to check that the TDRE flag is
before writing transmit data to TDR.

Operation when Multiple Receive Errors Occur Simultaneously

If a number of receive errors occur at the same time, the state of the status flags in SSI
shown in table 15.14. If there is an overrun error, data is not transferred from RSR to R
the receive data is lost.

Table 15.14 State of SSR Status Flags and Transfer of Receive Data

SSR Status Flags				Receive Data Transfer	
RDRF	ORER	FER	PER	RSR to RDR	Receive Errors
1	1	0	0	X	Overrun error
0	0	1	0	O	Framing error
0	0	0	1	O	Parity error
1	1	1	0	X	Overrun error + fra
1	1	0	1	X	Overrun error + par
0	0	1	1	O	Framing error + par
1	1	1	1	X	Overrun error + fra parity error

Notes: O: Receive data is transferred from RSR to RDR.
X: Receive data is not transferred from RSR to RDR.

Sending a Break

The TxD pin has a dual function as an I/O port whose direction (input or output) is dete
DR and DDR. This feature can be used to send a break.

Between serial transmission initialization and setting of the TE bit to 1, the mark state i
by the value of DR (the pin does not function as the TxD pin until the TE bit is set to 1)
Consequently, DDR and DR for the port corresponding to the TxD pin should first be s

To send a break during serial transmission, first clear DR to 0, then clear the TE bit to 0

When the TE bit is cleared to 0, the transmitter is initialized regardless of the current tr
state, the TxD pin becomes an I/O port, and 0 is output from the TxD pin.

Receive Error Flags and Transmit Operations (Synchronous Mode Only)

Transmission cannot be started when a receive error flag (ORER, PER, or FER) is set t
the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before startin
transmission.

Note also that receive error flags cannot be cleared to 0 even if the RE bit is cleared to 0

Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times t
rate.

In reception, the SCI samples the falling edge of the start bit using the base clock, and p
internal synchronization. Receive data is latched internally at the rising edge of the 8th
base clock. This is illustrated in figure 15.23.

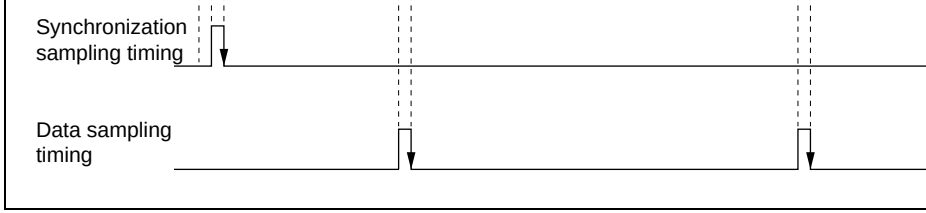


Figure 15.23 Receive Data Sampling Timing in Asynchronous Mode

Thus the receive margin in asynchronous mode is given by equation (1) below.

$$M = \left|\left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\right| \times 100\% \quad \text{..... (1)}$$

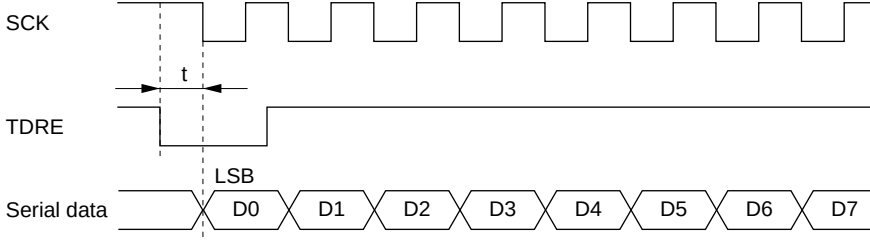
Where M: Receive margin (%)
N: Ratio of bit rate to clock (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock rate deviation

Assuming values of F = 0 and D = 0.5 in equation (1), a receive margin of 46.875% is equation (2) below.

When D = 0.5 and F = 0,

$$M = \left(0.5 - \frac{1}{2 \times 16}\right) \times 100\%$$
$$= 46.875\% \quad \text{..... (2)}$$

However, this is only a theoretical value, and a margin of 20% to 30% should be allow system design.



Note: When operating on an external clock, set t > 4 states.

Figure 15.24 Example of Synchronous Transmission by DTC

Examples: HD6432147SWFA

2. The product number is identical for F-ZTAT versions. However, be sure to inform Renesas sales representative if you will be using this option.

16.1 Overview

A two-channel I²C bus interface is available for the H8S/2148 Group and H8S/2147N option. The I²C bus interface conforms to and provides a subset of the Philips I²C bus bus) interface functions. The register configuration that controls the I²C bus differs pa Philips configuration, however.

Each I²C bus interface channel uses only one data line (SDA) and one clock line (SCL data, saving board and connector space.

16.1.1 Features

- Selection of addressing format or non-addressing format
  - I²C bus format: addressing format with acknowledge bit, for master/slave opera
  - Serial format: non-addressing format without acknowledge bit, for master oper
- 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 acknowledge output levels when receiving (I²C bus format)
- Automatic loading of acknowledge bit when transmitting (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 acknowledgement. The wait can be cleared by clearing the interrupt flag.

— Stop condition detection

- Selection of 16 internal clocks (in master mode)
- Direct bus drive (with SCL and SDA pins)
  — Two pins—P52/SCL0 and P97/SDA0—(normally NMOS push-pull outputs) fu
  NMOS open-drain outputs when the bus drive function is selected.
  — Two pins—P86/SCL1 and P42/SDA1—(normally CMOS pins) function as NM
  outputs when the bus drive function is selected.
- Automatic switching from formatless mode to I^2C bus format (channel 0 only)
  — Formatless operation (no start/stop conditions, non-addressing mode) in slave m
  — Operation using a common data pin (SDA) and independent clock pins (VSYN
  — Automatic switching from formatless mode to I^2C bus format on the fall of the S

16.1.2 Block Diagram

Figure 16.1 shows a block diagram of the I^2C bus interface.

Figure 16.2 shows an example of I/O pin connections to external circuits. Channel 0 I/O
channel 1 I/O pins differ in structure, and have different specifications for permissible a
voltages. For details, see section 26, Electrical Characteristics.

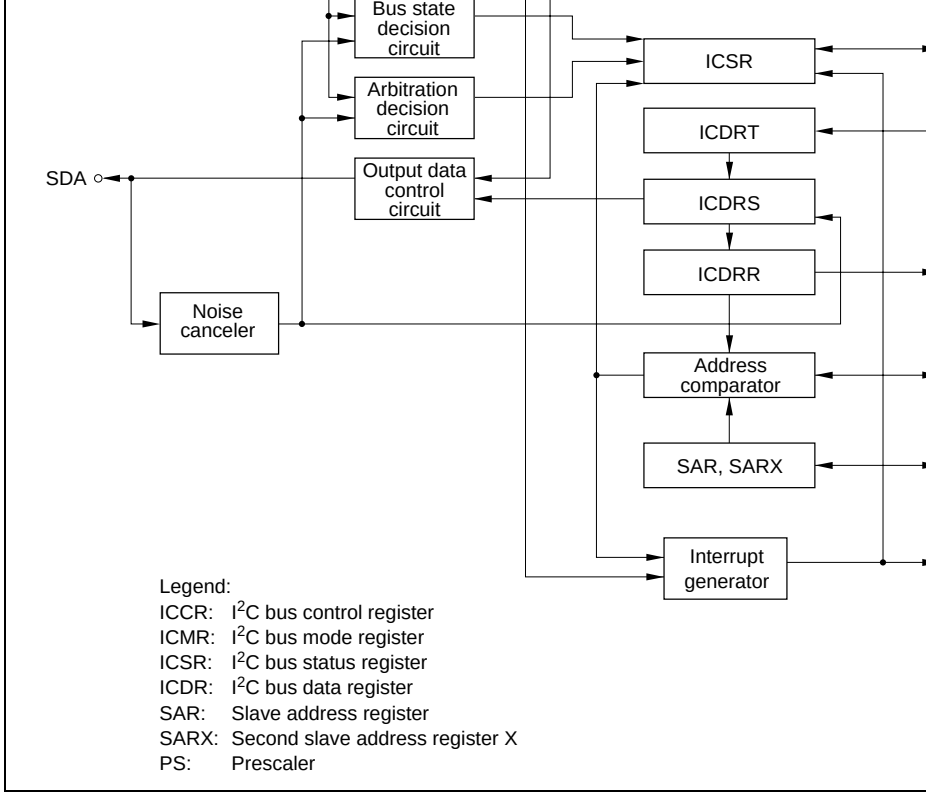


Figure 16.1 Block Diagram of I²C Bus Interface

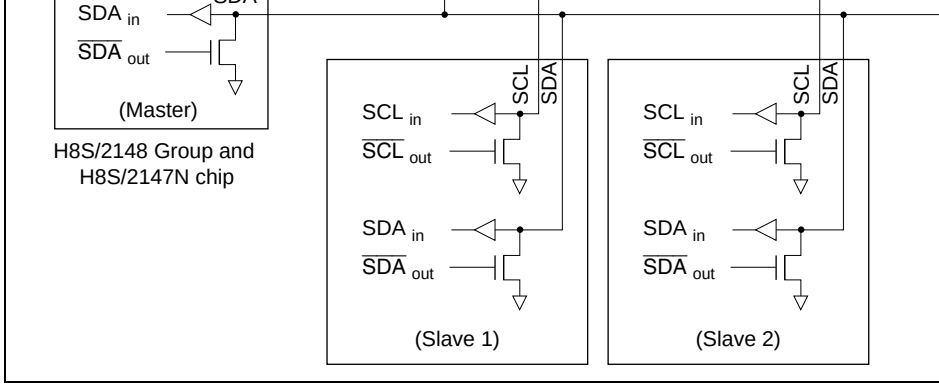


Figure 16.2 I²C Bus Interface Connections
(Example: H8S/2148 Group and H8S/2147N Chip as Master)

16.1.3 Input/Output Pins

Table 16.1 summarizes the input/output pins used by the I²C bus interface.

Table 16.1 I²C Bus Interface Pins

Channel	Name	Abbreviation*	I/O	Function
0	Serial clock	SCL0	I/O	IIC0 serial clock inpu
	Serial data	SDA0	I/O	IIC0 serial data input
	Formatless serial clock	VSYNCI	Input	IIC0 formatless seria
1	Serial clock	SCL1	I/O	IIC1 serial clock inpu
	Serial data	SDA1	I/O	IIC1 serial data input

Note: * In the text, the channel subscript is omitted, and only SCL and SDA are used

	I^2C bus data register	ICDR0	R/W	—	H
	I^2C bus mode register	ICMR0	R/W	H'00	H
	Slave address register	SAR0	R/W	H'00	H
	Second slave address register	SARX0	R/W	H'01	H
1	I^2C bus control register	ICCR1	R/W	H'01	H
	I^2C bus status register	ICSR1	R/W	H'00	H
	I^2C bus data register	ICDR1	R/W	—	H
	I^2C bus mode register	ICMR1	R/W	H'00	H
	Slave address register	SAR1	R/W	H'00	H
	Second slave address register	SARX1	R/W	H'01	H
Common	Serial/timer control register	STCR	R/W	H'00	H
	DDC switch register	DDCSWR	R/W	H'0F	H
	Module stop control register	MSTPCRH	R/W	H'3F	H
		MSTPCRL	R/W	H'FF	H

Notes: 1. Lower 16 bits of the address.

2. The register that can be written or read depends on the ICE bit in the I^2C bu register. The slave address register can be accessed when ICE = 0, and th mode register can be accessed when ICE = 1.

The I^2C bus interface registers are assigned to the same addresses as othe Register selection is performed by means of the IICE bit in the serial/timer c register (STCR).

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ICDRR

Bit	7	6	5	4	3	2	1
	ICDRR7	ICDRR6	ICDRR5	ICDRR4	ICDRR3	ICDRR2	ICDRR1
Initial value	—	—	—	—	—	—	—
Read/Write	R	R	R	R	R	R	R

- ICDRS

Bit	7	6	5	4	3	2	1
	ICDRS7	ICDRS6	ICDRR5	ICDRS4	ICDRS3	ICDRS2	ICDRS1
Initial value	—	—	—	—	—	—	—
Read/Write	—	—	—	—	—	—	—

- ICDRT

Bit	7	6	5	4	3	2	1
	ICDRT7	ICDRT6	ICDRT5	ICDRT4	ICDRT3	ICDRT2	ICDRT1
Initial value	—	—	—	—	—	—	—
Read/Write	W	W	W	W	W	W	W

- TDRE, RDRF (internal flags)

Bit							—
							TDRE
Initial value							0
Read/Write							—

ICDRT to ICDRS. If IIC is in receive mode and no previous data remains in ICDRR (
flag is 0) following transmission/reception of one frame of data using ICDRS, data is
automatically from ICDRS to ICDRR.

If the number of bits in a frame, excluding the acknowledge bit, is less than 8, transmi
receive data are stored differently. Transmit data should be written justified toward th
when MLS = 0, and toward the LSB side when MLS = 1. Receive data bits read from
side should be treated as valid when MLS = 0, and bits read from the MSB side when

ICDR is assigned to the same address as SARX, and can be written and read only whe
bit is set to 1 in ICCR.

The value of ICDR is undefined after a reset.

The TDRE and RDRF flags are set and cleared under the conditions shown below. Se
TDRE and RDRF flags affects the status of the interrupt flags.

- In receive mode (TRS = 0)

  (A 0 write to TRS during transfer is valid after reception of a frame conta
  acknowledge bit)

1	The next transmit data can be written in ICDR (ICDRT) [Setting conditions] • In transmit mode (TRS = 1), when a start condition is detected in the bu after a start condition is issued in master mode with the I^2C bus format c format selected • At the first transmit mode setting (TRS = 1) (first transmit mode setting c the mode is switched from I^2C bus mode to formatless mode • When data is transferred from ICDRT to ICDRS (Data transfer from ICDRT to ICDRS when TRS = 1 and TDRE = 0, and empty) • When detecting a start condition and then switching from slave receive = 0) state to transmit mode (TRS = 1) (first transmit mode switching only

RDRF	Description
0	The data in ICDR (ICDRR) is invalid (In [Clearing condition] When ICDR (ICDRR) receive data is read in receive mode
1	The ICDR (ICDRR) receive data can be read [Setting condition] When data is transferred from ICDRS to ICDRR (Data transfer from ICDRS to ICDRR in case of normal termination with TR RDRF = 0)

communication format. When the chip is in slave mode (and the addressing format is s
the upper 7 bits of SAR match the upper 7 bits of the first frame received after a start c
the chip operates as the slave device specified by the master device. SAR is assigned t
address as ICMR, and can be written and read only when the ICE bit is cleared to 0 in

SAR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 1—Slave Address (SVA6 to SVA0): Set a unique address in bits SVA6 to S
differing from the addresses of other slave devices connected to the I^2C bus.

Bit 0—Format Select (FS): Used together with the FSX bit in SARX and the SW bit
DDCSWR to select the communication format.

- I^2C bus format: addressing format with acknowledge bit
- Synchronous serial format: non-addressing format without acknowledge bit, for m
  only
- Formatless mode (channel 0 only): non-addressing format with or without acknow
  slave mode only, start/stop conditions not detected

The FS bit also specifies whether or not SAR slave address recognition is performed i
mode.

			• SARX slave address ignored
	1	0	I²C bus format • SAR slave address ignored • SARX slave address recognized
		1	Synchronous serial format • SAR and SARX slave addresses ignored
1	0	0	Formatless mode (start/stop conditions not detec
	0	1	• Acknowledge bit used
	1	0	
	1	1	Formatless mode* (start/stop conditions not dete • No acknowledge bit

Note: * Do not set this mode when automatic switching to the I²C bus format is perfo means of the DDCSWR setting.

16.2.3 Second Slave Address Register (SARX)

Bit	7	6	5	4	3	2	1	
	SVAX6	SVAX5	SVAX4	SVAX3	SVAX2	SVAX1	SVAX0	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

SARX is an 8-bit readable/writable register that stores the second slave address and sel communication format. When the chip is in slave mode (and the addressing format is se the upper 7 bits of SARX match the upper 7 bits of the first frame received after a start the chip operates as the slave device specified by the master device. SARX is assigned address as ICDR, and can be written and read only when the ICE bit is cleared to 0 in I

SARX is initialized to H'01 by a reset and in hardware standby mode.

- Formatless mode: non-addressing format with or without acknowledge bit, slave m start/stop conditions not detected

The FSX bit also specifies whether or not SARX slave address recognition is perform mode. For details, see the description of the FS bit in SAR.

16.2.4 I²C Bus Mode Register (ICMR)

Bit	7	6	5	4	3	2	1
	MLS	WAIT	CKS2	CKS1	CKS0	BC2	BC1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ICMR is an 8-bit readable/writable register that selects whether the MSB or LSB is tra first, performs master mode wait control, and selects the master mode transfer clock fr the transfer bit count. ICMR is assigned to the same address as SAR. ICMR can be wr read only when the ICE bit is set to 1 in ICCR.

ICMR is initialized to H'00 by a reset and in hardware standby mode.

Bit 7—MSB-First/LSB-First Select (MLS): Selects whether data is transferred MSB LSB-first.

If the number of bits in a frame, excluding the acknowledge bit, is less than 8, transmi receive data are stored differently. Transmit data should be written justified toward th when MLS = 0, and toward the LSB side when MLS = 1. Receive data bits read from side should be treated as valid when MLS = 0, and bits read from the MSB side when

Do not set this bit to 1 when the I²C bus format is used.

fall of the clock for the final data bit, the IRIC flag is set to 1 in ICCR, and a wait state
(with SCL at the low level). When the IRIC flag is cleared to 0 in ICCR, the wait ends
acknowledge bit is transferred. If WAIT is cleared to 0, data and acknowledge bits are
consecutively with no wait inserted.

The IRIC flag in ICCR is set to 1 on completion of the acknowledge bit transfer, regard
WAIT setting.

The setting of this bit is invalid in slave mode.

Bit 6

WAIT	Description	
0	Data and acknowledge bits transferred consecutively	(Initi
1	Wait inserted between data and acknowledge bits	

0	0	0	0	φ/28	179 kHz	286 kHz	357 kHz	571 kHz
			1	φ/40	125 kHz	200 kHz	250 kHz	400 kHz
		1	0	φ/48	104 kHz	167 kHz	208 kHz	333 kHz
			1	φ/64	78.1 kHz	125 kHz	156 kHz	250 kHz
	1	0	0	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz
			1	φ/100	50.0 kHz	80.0 kHz	100 kHz	160 kHz
		1	0	φ/112	44.6 kHz	71.4 kHz	89.3 kHz	143 kHz
			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	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz
		1	0	φ/96	52.1 kHz	83.3 kHz	104 kHz	167 kHz
			1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz
	1	0	0	φ/160	31.3 kHz	50.0 kHz	62.5 kHz	100 kHz
			1	φ/200	25.0 kHz	40.0 kHz	50.0 kHz	80.0 kHz
		1	0	φ/224	22.3 kHz	35.7 kHz	44.6 kHz	71.4 kHz
			1	φ/256	19.5 kHz	31.3 kHz	39.1 kHz	62.5 kHz

Note: * Outside the I^2C bus interface specification range (normal mode: max. 100 k
speed mode: max. 400 kHz).

Bits 2 to 0—Bit Counter (BC2 to BC0): Bits BC2 to BC0 specify the number of bits
transferred next. With the I^2C bus format (when the FS bit in SAR or the FSX bit in S
the data is transferred with one addition acknowledge bit. Bit BC2 to BC0 settings sho
during an interval between transfer frames. If bits BC2 to BC0 are set to a value other
the setting should be made while the SCL line is low.

The bit counter is initialized to 000 by a reset and when a start condition is detected. T
returns to 000 at the end of a data transfer, including the acknowledge bit.

	1	5	6	
1	0	6	7	
	1	7	8	

16.2.5 I²C Bus Control Register (ICCR)

Bit	7	6	5	4	3	2	1
	ICE	IEIC	MST	TRS	ACKE	BBSY	IRIC
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*

Note: * Only 0 can be written, to clear the flag.

ICCR is an 8-bit readable/writable register that enables or disables the I²C bus interface disables interrupts, selects master or slave mode and transmission or reception, enables acknowledgement, confirms the I²C bus interface bus status, issues start/stop conditions performs interrupt flag confirmation.

ICCR is initialized to H'01 by a reset and in hardware standby mode.

Bit 7—I²C Bus Interface Enable (ICE): Selects whether or not the I²C bus interface is used. When ICE is set to 1, port pins function as SCL and SDA input/output pins and tr operations are enabled. When ICE is cleared to 0, the I²C bus interface module is halted internal states are cleared.

The SAR and SARX registers can be accessed when ICE is 0. The ICMR and ICDR reg be accessed when ICE is 1.

Bit 6—I²C Bus Interface Interrupt Enable (IEIC): Enables or disables interrupts fr
bus interface to the CPU.

Bit 6	
IEIC	**Description**
0	Interrupts disabled
1	Interrupts enabled

Bit 5—Master/Slave Select (MST)
Bit 4—Transmit/Receive Select (TRS)

MST selects whether the I²C bus interface operates in master mode or slave mode.

TRS selects whether the I²C bus interface operates in transmit mode or receive mode.

In master mode with the I²C bus format, when arbitration is lost, MST and TRS are bo
hardware, causing a transition to slave receive mode. In slave receive mode with the a
format (FS = 0 or FSX = 0), hardware automatically selects transmit or receive mode
the R/$\overline{W}$ bit in the first frame after a start condition.

Modification of the TRS bit during transfer is deferred until transfer of the frame cont
acknowledge bit is completed, and the changeover is made after completion of the tra

MST and TRS select the operating mode as follows.

Bit 5

MST	Description
0	Slave mode (Ini [Clearing conditions] 1. When 0 is written by software 2. When bus arbitration is lost after transmission is started in I^2C bus forma mode
1	Master mode [Setting conditions] 1. When 1 is written by software (in cases other than clearing condition 2) 2. When 1 is written in MST after reading MST = 0 (in case of clearing con

Bit 4

TRS	Description
0	Receive mode (Ini [Clearing conditions] 1. When 0 is written by software (in cases other than setting condition 3) 2. When 0 is written in TRS after reading TRS = 1 (in case of clearing con 3. When bus arbitration is lost after transmission is started in I^2C bus forma mode 4. When the SW bit in DDCSWR changes from 1 to 0
1	Transmit mode [Setting conditions] 1. When 1 is written by software (in cases other than clearing conditions 3 2. When 1 is written in TRS after reading TRS = 0 (in case of clearing con and 4) 3. When a 1 is received as the R/$\overline{W}$ bit of the first frame in I^2C bus format s

other being IRIC). When the ACKE bit is 0, the TDRE, IRIC, and IRTR flags are set
completion of data transmission, regardless of the value of the acknowledge bit. When
bit is 1, the TDRE, IRIC, and IRTR flags are set on completion of data transmission w
acknowledge bit is 0, and the IRIC flag alone is set on completion of data transmissio
acknowledge bit is 1.

When the DTC is activated, the TDRE, IRIC, and IRTR flags are cleared to 0 after the
number of data transfers have been executed. Consequently, interrupts are not generat
continuous data transfer, but if data transmission is completed with a 1 acknowledge b
ACKE bit is set to 1, the DTC is not activated and an interrupt is generated, if enabled

Depending on the receiving device, the acknowledge bit may be significant, in indicat
completion of processing of the received data, for instance, or may be fixed at 1 and h
significance.

Bit 3

ACKE	Description
0	The value of the acknowledge bit is ignored, and continuous transfer (is performed
1	If the acknowledge bit is 1, continuous transfer is interrupted

Bit 2—Bus Busy (BBSY): The BBSY flag can be read to check whether the I^2C bus (
is busy or free. In master mode, this bit is also used to issue start and stop conditions.

A high-to-low transition of SDA while SCL is high is recognized as a start condition,
BBSY to 1. A low-to-high transition of SDA while SCL is high is recognized as a stop
clearing BBSY to 0.

To issue a start condition, use a MOV instruction to write 1 in BBSY and 0 in SCP. A
start condition is issued in the same way. To issue a stop condition, use a MOV instru
write 0 in BBSY and 0 in SCP. It is not possible to write to BBSY in slave mode; the
interface must be set to master transmit mode before issuing a start condition. MST an
should both be set to 1 before writing 1 in BBSY and 0 in SCP.

Bit 1—I²C Bus Interface Interrupt Request Flag (IRIC): Indicates that the I²C bus i
issued an interrupt request to the CPU. IRIC is set to 1 at the end of a data transfer, whe
address or general call address is detected in slave receive mode, when bus arbitration i
master transmit mode, and when a stop condition is detected. IRIC is set at different tin
depending on the FS bit in SAR and the WAIT bit in ICMR. See section 16.3.6, IRIC S
Timing and SCL Control. The conditions under which IRIC is set also differ depending
setting of the ACKE bit in ICCR.

IRIC is cleared by reading IRIC after it has been set to 1, then writing 0 in IRIC.

When the DTC is used, IRIC is cleared automatically and transfer can be performed co
without CPU intervention.

[Setting conditions]

- I^2C bus format master mode
  1. When a start condition is detected in the bus line state after a start condition
     (when the TDRE flag is set to 1 because of first frame transmission)
  2. When a wait is inserted between the data and acknowledge bit when WAIT =
  3. At the end of data transfer
     (at the rise of the 9th transmit/receive clock pulse when no wait is inserted, (
     when a wait is inserted (WAIT=1), at the fall of the 8th transmit/receive clock
  4. When a slave address is received after bus arbitration is lost
     (when the AL flag is set to 1)
  5. When 1 is received as the acknowledge bit when the ACKE bit is 1
     (when the ACKB bit is set to 1)
- I^2C bus format slave mode
  1. When the slave address (SVA, SVAX) matches
     (when the AAS and AASX flags are set to 1)
     and at the end of data transfer up to the subsequent retransmission start cor
     condition detection
     (when the TDRE or RDRF flag is set to 1)
  2. When the general call address is detected
     (when FS = 0 and the ADZ flag is set to 1)
     and at the end of data transfer up to the subsequent retransmission start cor
     condition detection
     (when the TDRE or RDRF flag is set to 1)
  3. When 1 is received as the acknowledge bit when the ACKE bit is 1
     (when the ACKB bit is set to 1)
  4. When a stop condition is detected
     (when the STOP or ESTP flag is set to 1)
- Synchronous serial format, and formatless mode
  1. At the end of data transfer
     (when the TDRE or RDRF flag is set to 1)
  2. When a start condition is detected with serial format selected
  3. When the SW bit is set to 1 in DDCSWR

Except the above, when the conditions to set the TDRE or RDRF internal flag to 1 i

Even when the IRIC flag and IRTR flag are set, the TDRE or RDRF internal flag may n
The IRIC and IRTR flags are not cleared at the end of the specified number of transfers
continuous transfer using the DTC. The TDRE or RDRF flag is cleared, however, since
specified number of ICDR reads or writes have been completed.

Table 16.3 shows the relationship between the flags and the transfer states.

Table 16.3 Flags and Transfer States

MST	TRS	BBSY	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB	State
1/0	1/0	0	0	0	0	0	0	0	0	0	Idle state (flag required)
1	1	0	0	0	0	0	0	0	0	0	Start condition
1	1	1	0	0	1	0	0	0	0	0	Start condition
1	1/0	1	0	0	0	0	0	0	0	0/1	Master mode w
1	1/0	1	0	0	1	0	0	0	0	0/1	Master mode t receive end
0	0	1	0	0	0	1/0	1	1/0	1/0	0	Arbitration lost
0	0	1	0	0	0	0	0	1	0	0	SAR match by slave mode
0	0	1	0	0	0	0	0	1	1	0	General call ac
0	0	1	0	0	0	1	0	0	0	0	SARX match
0	1/0	1	0	0	0	0	0	0	0	0/1	Slave mode transmit/receiv (except after S
0	1/0	1	0	0	1	1	0	0	0	0	Slave mode
0	1	1	0	0	0	1	0	0	0	1	transmit/receiv SARX match)
0	1/0	0	1/0	1/0	0	0	0	0	0	0/1	Stop condition

1	Reading always returns a value of 1 Writing is ignored	(I

16.2.6 I²C Bus Status Register (ICSR)

Bit	7	6	5	4	3	2	1
	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * Only 0 can be written, to clear the flags.

ICSR is an 8-bit readable/writable register that performs flag confirmation and acknov
confirmation and control.

ICSR is initialized to H'00 by a reset and in hardware standby mode.

	2. When the IRIC flag is cleared to 0
1	• In I²C bus format slave mode Error stop condition detected [Setting condition] When a stop condition is detected during frame transfer • In other modes No meaning

Bit 6—Normal Stop Condition Detection Flag (STOP): Indicates that a stop conditio
detected after completion of frame transfer in I²C bus format slave mode.

Bit 6

STOP	Description
0	No normal stop condition (Ini [Clearing conditions] 1. When 0 is written in STOP after reading STOP = 1 2. When the IRIC flag is cleared to 0
1	• In I²C bus format slave mode Normal stop condition detected [Setting condition] When a stop condition is detected after completion of frame transfer • In other modes No meaning

**Bit 5—I²C Bus Interface Continuous Transmission/Reception Interrupt Request F
(IRTR):** Indicates that the I²C bus interface has issued an interrupt request to the CPU,
source is completion of reception/transmission of one frame in continuous transmissio
for which DTC activation is possible. When the IRTR flag is set to 1, the IRIC flag is a
at the same time.

	1. When 0 is written in IRTR after reading IRTR = 1 2. When the IRIC flag is cleared to 0
1	Continuous transfer state [Setting conditions] • In I^2C bus interface slave mode When the TDRE or RDRF flag is set to 1 when AASX = 1 • In other modes When the TDRE or RDRF flag is set to 1

Bit 4—Second Slave Address Recognition Flag (AASX): In I^2C bus format slave re this flag is set to 1 if the first frame following a start condition matches bits SVAX6 to SARX.

AASX is cleared by reading AASX after it has been set to 1, then writing 0 in AASX. also cleared automatically when a start condition is detected.

Bit 4

AASX	Description
0	Second slave address not recognized (I [Clearing conditions] 1. When 0 is written in AASX after reading AASX = 1 2. When a start condition is detected 3. In master mode
1	Second slave address recognized [Setting condition] When the second slave address is detected in slave receive mode while F

Bit 3

AL	Description
0	Bus arbitration won (Ini [Clearing conditions] 1. When ICDR data is written (transmit mode) or read (receive mode) 2. When 0 is written in AL after reading AL = 1
1	Arbitration lost [Setting conditions] 1. If the internal SDA and SDA pin disagree at the rise of SCL in master tra mode 2. If the internal SCL line is high at the fall of SCL in master transmit mode

Bit 2—Slave Address Recognition Flag (AAS): In I^2C bus format slave receive mode set to 1 if the first frame following a start condition matches bits SVA6 to SVA0 in SA general call address (H'00) is detected.

AAS is cleared by reading AAS after it has been set to 1, then writing 0 in AAS. In add is reset automatically by write access to ICDR in transmit mode, or read access to ICDI mode.

1	Slave address or general call address recognized [Setting condition] When the slave address or general call address is detected in slave receive FS = 0

Bit 1—General Call Address Recognition Flag (ADZ): In I^2C bus format slave rece
this flag is set to 1 if the first frame following a start condition is the general call addr

ADZ is cleared by reading ADZ after it has been set to 1, then writing 0 in ADZ. In ac
is reset automatically by write access to ICDR in transmit mode, or read access to ICD
mode.

Bit 1

ADZ	Description
0	General call address not recognized [Clearing conditions] 1. When ICDR data is written (transmit mode) or read (receive mode) 2. When 0 is written in ADZ after reading ADZ = 1 3. In master mode
1	General call address recognized [Setting condition] When the general call address is detected in slave receive mode while FSX

When this bit is written to, the acknowledge data transmitted at the receipt is rewritten
of the TRS value. The data loaded fom the receiving device is retained, therefore take c
using bit-manipulation instructions.

Bit 0

ACKB	Description
0	Receive mode: 0 is output at acknowledge output timing (I Transmit mode: Indicates that the receiving device has acknowledged the d is 0)
1	Receive mode: 1 is output at acknowledge output timing Transmit mode: Indicates that the receiving device has not acknowledged t (signal is 1)

16.2.7 Serial/Timer Control Register (STCR)

Bit	7	6	5	4	3	2	1
	IICS	IICX1	IICX0	IICE	FLSHE	—	ICKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

STCR is an 8-bit readable/writable register that controls register access, the I^2C interfac
mode (when the on-chip IIC option is included), and on-chip flash memory (F-ZTAT v
and selects the TCNT input clock source. For details of functions not related to the I^2C
interface, see section 3.2.4, Serial Timer Control Register (STCR), and the descriptions
relevant modules. If a module controlled by STCR is not used, do not write 1 to the cor
bit.

STCR is initialized to H'00 by a reset and in hardware standby mode.

Bits 6 and 5—I²C Transfer Select 1 and 0 (IICX1 and 0): This bit, together with bi
CKS0 in ICMR, selects the transfer rate in master mode. For details, see section 16.2.
Mode Register (ICMR).

Bit 4—I²C Master Enable (IICE): Controls CPU access to the I²C bus interface data
registers (ICCR, ICSR, ICDR/SARX, ICMR/SAR).

Bit 4

IICE	Description	
0	CPU access to I²C bus interface data and control registers is disabled	(
1	CPU access to I²C bus interface data and control registers is enabled	

Bit 3—Flash Memory Control Register Enable (FLSHE): Controls CPU access to
memory control registers, the power-down mode control registers, and the supporting
control registers. See section 3.2.4, Serial Timer Control Register (STCR), for details.

Bit 2—Reserved: Do not write 1 to this bit.

Bits 1 and 0—Internal Clock Source Select 1 and 0 (ICKS1, ICSK0): These bits, t
bits CKS2 to CKS0 in TCR, select the clock input to the timer counters (TCNT). For
section 12.2.4, Timer Control Register (TCR).

2. Always read as 1.

DDCSWR is an 8-bit readable/writable register that is used to initialize IIC and control internal latch clearance.

DDCSWR is initialized to H'0F by a reset and in hardware standby mode.

Bit 7—DDC Mode Switch Enable (SWE): Selects the function for automatically swit channel 0 from formatless mode to the I^2C bus format.

Bit 7

SWE	Description
0	Automatic switching of IIC channel 0 from formatless mode to I^2C bus forma disabled (In
1	Automatic switching of IIC channel 0 from formatless mode to I^2C bus forma enabled

Bit 6—DDC Mode Switch (SW): Selects either formatless mode or the I^2C bus format channel 0.

Bit 6

SW	Description
0	IIC channel 0 is used with the I^2C bus format (Ini [Clearing conditions] 1. When 0 is written by software 2. When a falling edge is detected on the SCL pin when SWE = 1
1	IIC channel 0 is used in formatless mode [Setting condition] When 1 is written in SW after reading SW = 0

Bit 4—DDC Mode Switch Interrupt Flag (IF): Flag that indicates an interrupt requ
CPU when automatic format switching is executed for IIC channel 0.

Bit 4

IF	Description
0	No interrupt is requested when automatic format switching is executed [Clearing condition] When 0 is written in IF after reading IF = 1
1	An interrupt is requested when automatic format switching is executed [setting condition] When a falling edge is detected on the SCL pin when SWE = 1

Bits 3 to 0—IIC Clear 3 to 0 (CLR3 to CLR0): These bits control initialization of th
state of IIC0 and IIC1.

These bits can only be written to; if read they will always return a value of 1.

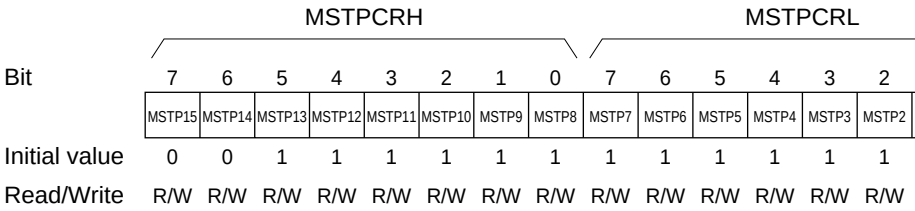
When a write operation is performed on these bits, a clear signal is generated for the i
circuit of the corresponding module(s), and the internal state of the IIC module(s) is in

The write data for these bits is not retained. To perform IIC clearance, bits CLR3 to C
written to simultaneously using an MOV instruction. Do not use a bit-manipulation in
such as BCLR.

When clearing is required again, all the bits must be written to in accordance with the

1	—	—	—	Invalid setting

16.2.9 Module Stop Control Register (MSTPCR)



MSTPCR comprises two 8-bit readable/writable registers, and is used to perform modu mode control.

When the MSTP4 or MSTP3 bit is set to 1, operation of the corresponding IIC channel the end of the bus cycle, and a transition is made to module stop mode. For details, see 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

MSTPCRL Bit 4—Module Stop (MSTP4): Specifies IIC channel 0 module stop mod

MSTPCRL Bit 4	
MSTP4	**Description**
0	IIC channel 0 module stop mode is cleared
1	IIC channel 0 module stop mode is set (I

16.3 Operation

16.3.1 I^2C Bus Data Format

The I^2C bus interface has serial and I^2C bus formats.

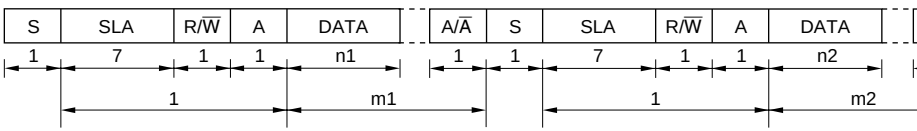
The I^2C bus formats are addressing formats with an acknowledge bit. These are shown
16.3 (a) and (b). The first frame following a start condition always consists of 8 bits.

IIC channel 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. Although start
conditions must be issued, this format can be used as a synchronous serial format. Thi
figure 16.5.

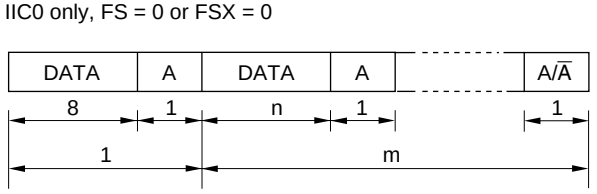
Figure 16.6 shows the I^2C bus timing.

The symbols used in figures 16.3 to 16.6 are explained in table 16.4.



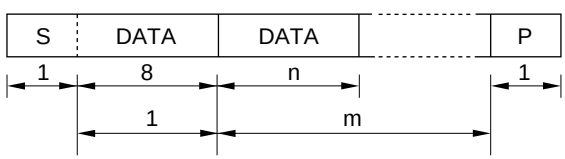
Legend:
n1 and n2: transfer bit count (n1 and n2 = 1 to 8
m1 and m2: transfer frame count (m1 and m2 ≥

Figure 16.3 I²C Bus Data Formats (I²C Bus Formats)



Legend:
n: transfer bit count (n =
m: transfer frame count (

Figure 16.4 Formatless



Legend:
n: transfer bit count
(n = 1 to 8)
m: transfer frame coun
(m ≥ 1)

Figure 16.5 I²C Bus Data Format (Serial Format)

Table 16.4 I²C Bus Data Format Symbols

Legend	
S	Start condition. The master device drives SDA from high to low while SCL
SLA	Slave address, by which the master device selects a slave device
R/$\overline{W}$	Indicates the direction of data transfer: from the slave device to the maste when R/$\overline{W}$ is 1, or from the master device to the slave device when R/$\overline{W}$ is
A	Acknowledge. The receiving device (the slave in master transmit mode, or in master receive mode) drives SDA low to acknowledge a transfer
DATA	Transferred data. The bit length is set by bits BC2 to BC0 in ICMR. The M LSB-first format is selected by bit MLS in ICMR
P	Stop condition. The master device drives SDA from low to high while SCL

16.3.2 Master Transmit Operation

In I²C bus format master transmit mode, the master device outputs the transmit clock a data, and the slave device returns an acknowledge signal.

The transmission procedure and operations by which data is sequentially transmitted i synchronization with ICDR write operations, are described below.

(1) Set the ICE bit in ICCR to l. Set bits MLS, WAIT, and CKS2 to CKS0 in ICMR, in STCR, according to the operation mode.
(2) Read the BBSY flag to confirm that the bus is free.
(3) Set the MST and TRS bits to 1 in ICCR to select master transmit mode.
(4) Write 1 to BBSY and 0 to SCP. This switches SDA from high to low when SCL i generates the start condition.
(5) When the start condition is generated, the IRIC and IRTR flags are set to 1. If the ICCR has been set to l, an interrupt request is sent to the CPU.

cleared, the end of transfer cannot be identified.

The master device sequentially sends the transmit clock and the data written to ICI timing shown in figure 16.7. The selected slave device (i.e., the slave device with t matching slave address) drives SDA low at the 9th transmit clock pulse and returns acknowledge signal.

(7) When one frame of data has been transmitted, the IRIC flag is set to 1 at the rise of transmit clock pulse. After one frame has been transmitted, SCL is automatically f synchronization with the internal clock until the next transmit data is written.

(8) Read the ACKB bit to confirm that ACKB is 0. When the slave device has not retu acknowledge signal and ACKB remains 1, execute the transmit end processing des step (12) and perform transmit operation again.

(9) Write the next data to be transmitted in ICDR. To indicate the end of data transfer, IRIC flag to 0.

As described in step (6) above, writing to ICDR and clearing of the IRIC flag must executed continuously so that no interrupt is inserted.

The next frame is transmitted 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 transmit clock pulse. After one frame has been transmitted, SCL is automatically f synchronization with the internal clock until the next transmit data is written.

(11) Read the ACKB bit of ICSR. Confirm that the slave device has returned an acknov signal and ACKB is 0. When more data is to be transmitted, return to step (9) to ex transmit operation. If the slave device has not returned an acknowledge signal and execute the transmit end processing described in step (12).

(12) Clear the IRIC flag to 0. Write BBSY and SCP of ICCR to 0. By doing so, SDA is from low to high while SCL is high and the transmit stop condition is generated.

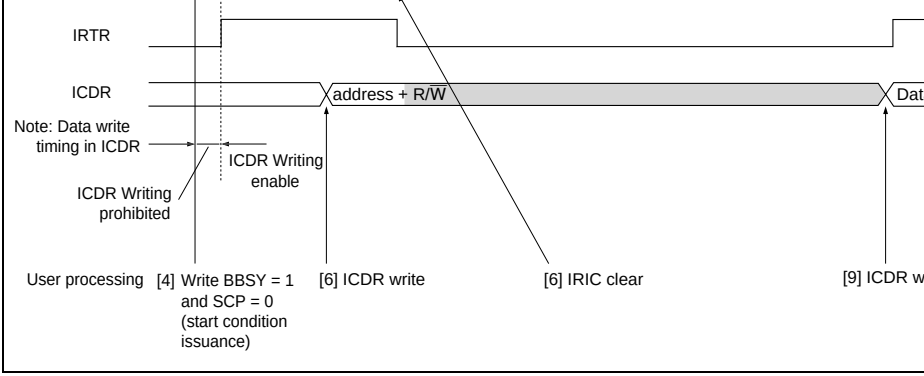


Figure 16.7 Example of Master Transmit Mode Operation Timing (MLS = WAIT = 0)

16.3.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, receives data, and acknowledge signal. The slave device transmits data.

The receive procedure and operations by which data is sequentially received in synchr with ICDR read operations, are described below.

(1) Clear the TRS bit of ICCR to 0 and switch from transmit mode to receive mode. S WAIT bit to 1 and clear the ACKB bit of ICSR to 0 (acknowledge data setting).

(2) When ICDR is read (dummy data read), reception is started and the receive clock and data is received, in synchronization with the internal clock. To indicate the w IRIC flag to 0.

Reading from ICDR and clearing of the IRIC f1ag must be executed continuously interrupt is inserted.

If a period of time that is equal to transfer one byte has elapsed by the time the IR cleared, the end of transfer cannot be identified.

(5) When one frame of data has been transmitted, the IRIC and IRTR flags are set to 1 of the 9th transmit clock pulse.

The master device continues to output the receive clock for the next receive data.

(6) Read the ICDR receive data.

(7) Clear the IRIC flag to indicate the next wait.

From clearing of the IRIC flag to completion of data transmission as described in s (6), and (7), must be performed within the time taken to transfer one byte, because of the wait state as described in step (4) (or (9)).

(8) The IRIC flag is set to 1 at the fall of the 8th one-frame reception clock pulse. SCL automatically fixed low in synchronization with the internal clock until the IRIC fl cleared. If this frame is the final reception frame, execute the end processing as des (l0).

(9) Clear the IRIC flag to 0 to release from the wait state. The master device outputs th reception clock pulse, sets SDA to low, and returns an acknowledge signal.

By repeating steps (5) to (9) above, more data can be received.

(l0) Set the ACKB bit of ICSR to 1 and set the acknowledge data for the final reception

Set the TRS bit of ICCR to 1 to change receive mode to transmit mode.

(11) Clear the IRIC flag to release from the wait state.

(12) When one frame of data has been received, the IRIC flag is set to 1 at the rise of th reception clock pulse.

(13) Clear the WAIT bit of ICMR to 0 to cancel wait mode. Read the ICDR receive dat the IRIC flag to 0.

Clear the IRIC flag only when WAIT = 0.

(If the stop-condition generation command is executed after clearing the IRIC flag then clearing the WAIT bit to 0, the SDA line is fixed low and the stop condition c generated.)

(14) Write 0 to BBSY and SCP. This changes SDA from low to high when SCL is high generates the stop condition.

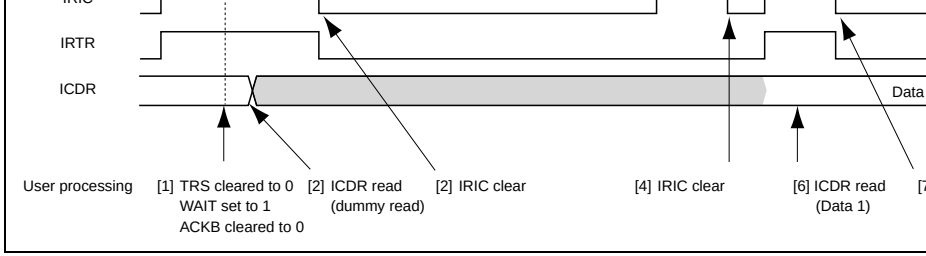


Figure 16.8 (a) Example of Master Receive Mode Operation Timing (MLS = ACKB = 0, WAIT = 1)

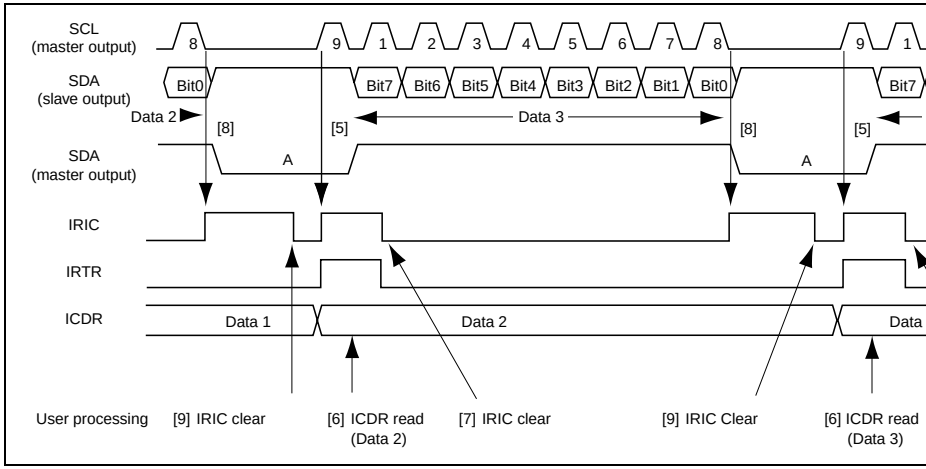


Figure 16.8 (b) Example of Master Receive Mode Operation Timing (MLS = ACKB = 0, WAIT = 1)

[2] When the start condition output by the master device is detected, the BBSY flag in I to 1.

[3] When the slave address matches in the first frame following the start condition, the operates as the slave device specified by the master device. If the 8th data bit (R/$\overline{W}$) TRS bit in ICCR remains cleared to 0, and slave receive operation is performed.

[4] At the 9th clock pulse of the receive frame, the slave device drives SDA low and re acknowledge signal. At the same time, the IRIC flag in ICCR is set to 1. If the IEIC ICCR has been set to 1, an interrupt request is sent to the CPU. If the RDRF interna been cleared to 0, it is set to 1, and the receive operation continues. If the RDRF int has been set to 1, the slave device drives SCL low from the fall of the receive clock is read into ICDR.

[5] Read ICDR and clear the IRIC flag in ICCR to 0. The RDRF flag is cleared to 0.

Receive operations can be performed continuously by repeating steps [4] and [5]. Wher changed from low to high when SCL is high, and the stop condition is detected, the BB ICCR is cleared to 0.

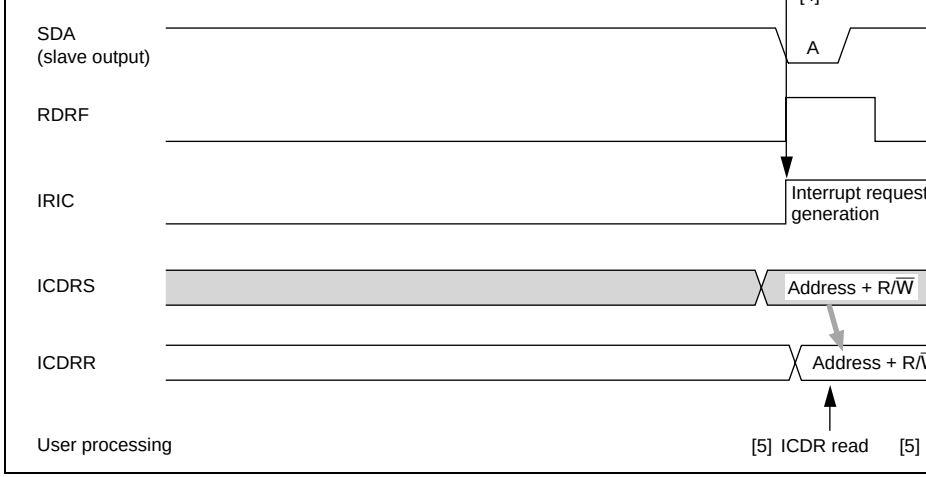


Figure 16.9 Example of Slave Receive Mode Operation Timing (1) (MLS = A

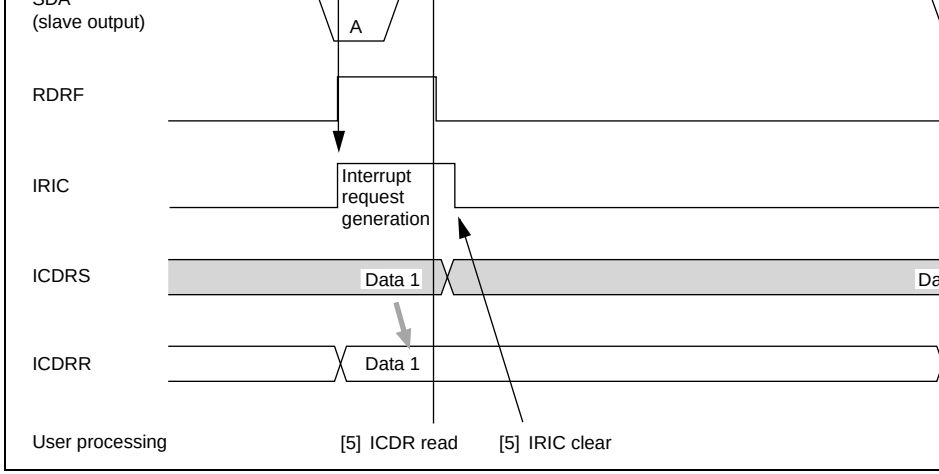


Figure 16.10 Example of Slave Receive Mode Operation Timing (2) (MLS = AC

[2] When the slave address matches in the first frame following detection of the start c
the slave device drives SDA low at the 9th clock pulse and returns an acknowledg
the same time, the IRIC flag in ICCR is set to 1. If the IEIC bit in ICCR has been s
interrupt request is sent to the CPU. If the 8th data bit (R/$\overline{W}$) is 1, the TRS bit in IC
1, and the mode changes to slave transmit mode automatically. The TDRE internal
to 1. The slave device drives SCL low from the fall of the transmit clock until ICD
written.

[3] After clearing the IRIC flag to 0, write data to ICDR. The TDRE internal flag is cl
The written data is transferred to ICDRS, and the TDRE internal flag and the IRIC
flags are set to 1 again. After clearing the IRIC flag to 0, write the next data to ICD
slave device sequentially sends the data written into ICDR in accordance with the
by the master device at the timing shown in figure 16.11.

[4] When one frame of data has been transmitted, the IRIC flag in ICCR is set to 1 at
the 9th transmit clock pulse. If the TDRE internal flag has been set to 1, this slave
drives SCL low from the fall of the transmit clock until data is written to ICDR. Th
device drives SDA low at the 9th clock pulse, and returns an acknowledge signal.
acknowledge signal is stored in the ACKB bit in ICSR, this bit can be used to dete
whether the transfer operation was performed normally. When the TDRE internal
data written into ICDR is transferred to ICDRS, transmission is started, and the TD
flag and the IRIC and IRTR flags are set to 1 again.

[5] To continue transmission, clear the IRIC flag to 0, then write the next data to be tr
into ICDR. The TDRE internal flag is cleared to 0.

Transmit operations can be performed continuously by repeating steps [4] and [5]. To
transmission, write H'FF to ICDR to release SDA on the slave side. When SDA is cha
low to high when SCL is high, and the stop condition is detected, the BBSY flag in IC
cleared to 0.

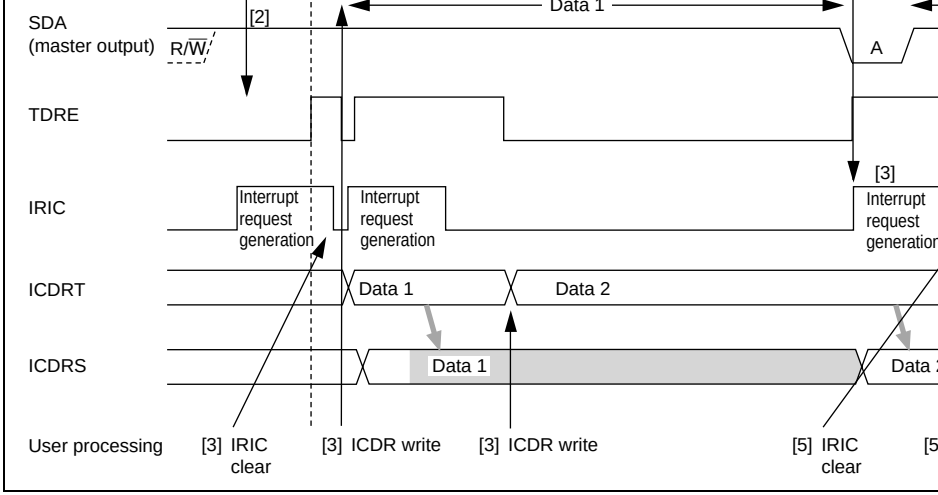
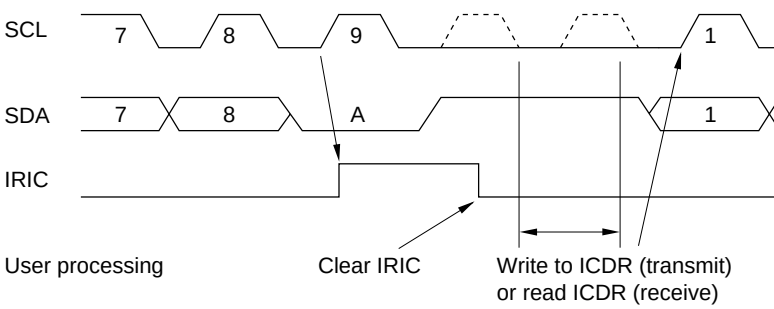
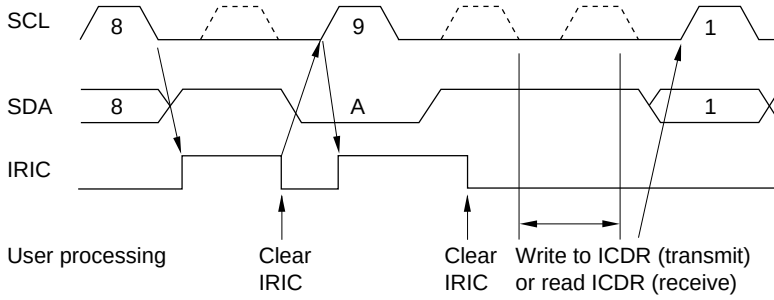


Figure 16.11 Example of Slave Transmit Mode Operation Timing (MLS =



(b) When WAIT = 1, and FS = 0 or FSX = 0 (I^2C bus format, wait inserted)



(c) When FS = 1 and FSX = 1 (synchronous serial format)

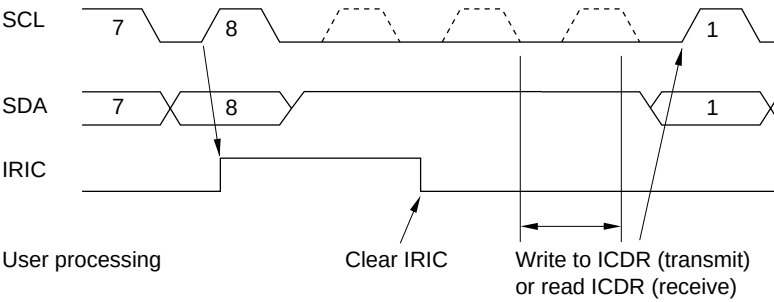


Figure 16.12 IRIC Setting Timing and SCL Control

- Separate clock pins for formatless operation (VSYNCI) and I²C bus format operatio
- A fixed 1 level for the SCL pin during formatless operation (the SCL pin does not c level)
- Settings of bits other than TRS in ICCR that allow I²C bus format operation

Automatic switching is performed from formatless mode to the I²C bus format when th DDCSWR is automatically cleared to 0 on detection of a falling edge on the SCL pin. S from the I²C bus format to formatless mode is achieved by having software set the SW DDCSWR to 1.

In formatless mode, bits (such as MSL and TRS) that control the I²C bus interface oper must not be modified. When switching from the I²C bus format to formatless mode, set bit to 1 or clear it to 0 according to the transmit data (transmission or reception) in form mode, then set the SW bit to 1. After automatic switching from formatless mode to the format (slave mode), in order to wait for slave address reception, the TRS bit is automa cleared to 0.

If a falling edge is detected on the SCL pin during formatless operation, the I²C bus inte operating mode is switched to the I²C bus format without waiting for a stop condition to detected.

number of transfer data bytes is known in slave mode.

Table 16.5 Examples of Operation Using the DTC

Item	Master Transmit Mode	Master Receive Mode	Slave Transmit Mode	Slave R Mode
Slave address + R/$\overline{W}$ bit transmission/ reception	Transmission by DTC (ICDR write)	Transmission by CPU (ICDR write)	Reception by CPU (ICDR read)	Recept (ICDR
Dummy data read	—	Processing by CPU (ICDR read)	—	—
Actual data transmission/ reception	Transmission by DTC (ICDR write)	Reception by DTC (ICDR read)	Transmission by DTC (ICDR write)	Recept (ICDR
Dummy data (H'FF) write	—	—	Processing by DTC (ICDR write)	—
Last frame processing	Not necessary	Reception by CPU (ICDR read)	Not necessary	Recept (ICDR
Transfer request processing after last frame processing	1st time: Clearing by CPU 2nd time: End condition issuance by CPU	Not necessary	Automatic clearing on detection of end condition during transmission of dummy data (H'FF)	Not nec
Setting of number of DTC transfer data frames	Transmission: Actual data count + 1 (+1 equivalent to slave address + R/$\overline{W}$ bits)	Reception: Actual data count	Transmission: Actual data count + 1 (+1 equivalent to dummy data (H'FF))	Recept data co

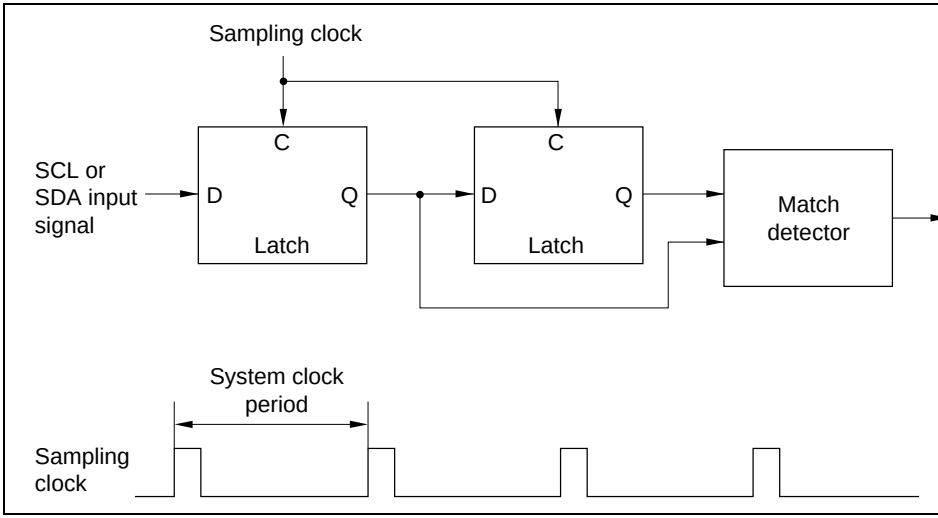


Figure 16.13 Block Diagram of Noise Canceler

16.3.10 Sample Flowcharts

Figures 16.14 to 16.17 show sample flowcharts for using the I^{2}C bus interface in each r

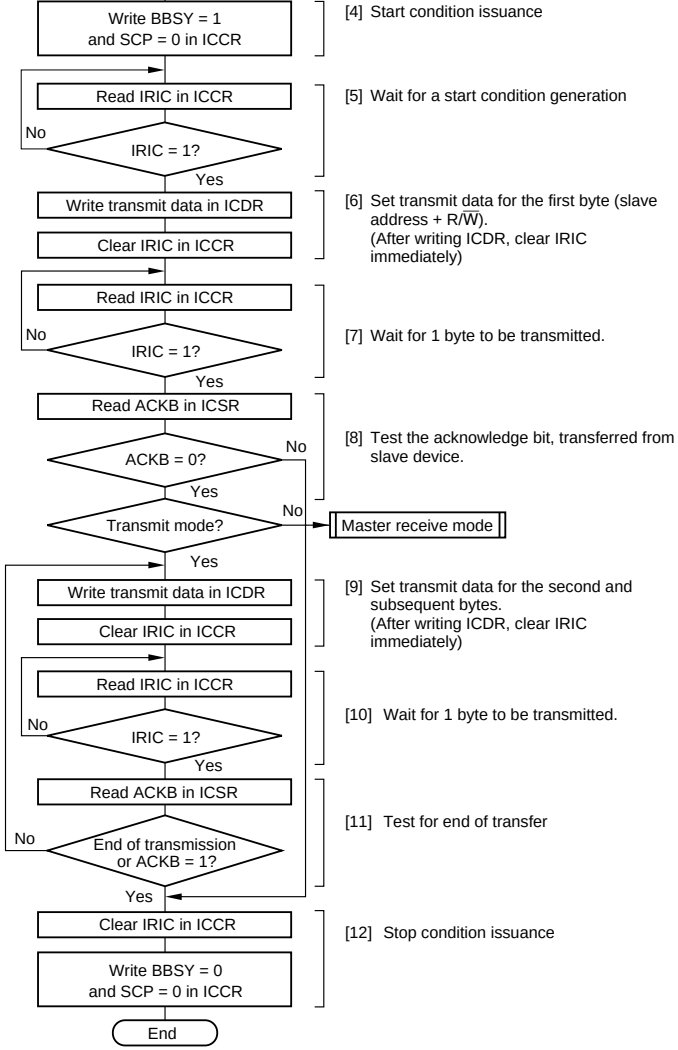


Figure 16.14 Flowchart for Master Transmit Mode (Example)

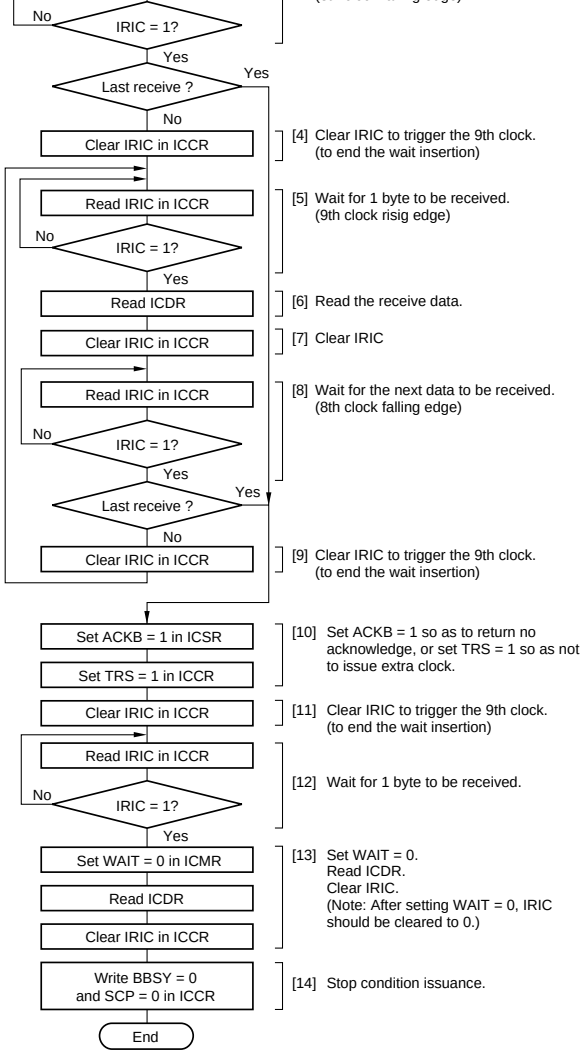


Figure 16.15 Flowchart for Master Receive Mode (Example)

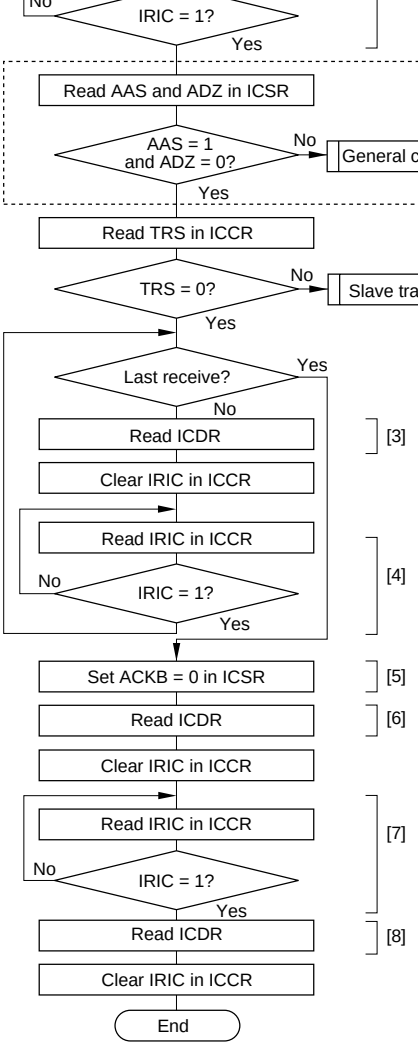
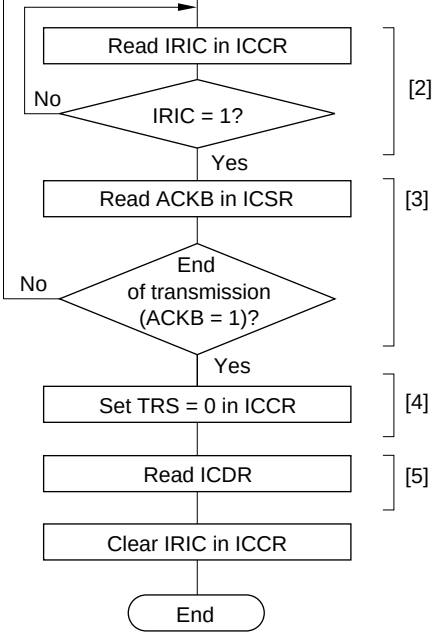


Figure 16.16 Flowchart for Slave Receive Mode (Example)



[4] Select slave receive mode.

[5] Dummy read (to release the SCL l

Figure 16.17 Flowchart for Slave Transmit Mode (Example)

Scope of Initialization:
The initialization executed by this function covers the following items:

- TDRE and RDRF 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, DDCSWR, STC
- Internal latches used to retain register read information for setting/clearing flags in ICCR, ICSR, and DDCSWR registers
- The value of the ICMR register bit counter (BC2 to BC0)
- Generated interrupt sources (interrupt sources transferred to the interrupt controlle

Notes on Initialization:
- Interrupt flags and interrupt sources are not cleared, and so flag clearing measures taken as necessary.
- Basically, other register flags are not cleared either, and so flag clearing measures taken as necessary.
- When initialization is performed by means of the DDCSWR register, the write dat CLR3 to CLR0 is not retained. To perform IIC clearance, bits CLR3 to CLR0 mus to simultaneously using an MOV instruction. Do not use a bit-manipulation instruc 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 wi transmitting/receiving at that point and the SCL and SDA pins will be released. W transmission/reception is started again, register initialization, etc., must be carried necessary to enable correct communication as a system.

bit.

2. Execute a stop condition issuance instruction (write 0 to BBSY and SCP) to clear th bit to 0, and wait for two transfer rate clock cycles.
3. Re-execute initialization of the internal state by setting of bit CLR3 to CLR0 or by ICE bit.
4. Initialize (re-set) the IIC registers.

16.4 Usage Notes

- In master mode, if an instruction to generate a start condition is immediately follow instruction to generate a stop condition, neither condition will be output correctly. T consecutive start and stop conditions, after issuing the instruction that generates the condition, read the relevant ports, check that SCL and SDA are both low, then issue instruction that generates the stop condition. Note that SCL may not yet have gone l BBSY is cleared to 0.

- Either of the following two conditions will start the next transfer. Pay attention to th conditions when reading or writing to ICDR.
  - Write access to ICDR when ICE = 1 and TRS = 1 (including automatic transfer ICDRT to ICDRS)
  - Read access to ICDR when ICE = 1 and TRS = 0 (including automatic transfer f ICDRS to ICDRR)

- Table 16.6 shows the timing of SCL and SDA output in synchronization with the in clock. Timings on the bus are determined by the rise and fall times of signals affect 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)		$1t_{SCLL} - (6t_{cyc}$ 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 input is sampled in synchronization with the internal clock. The AC therefore depends on the system clock cycle t_{cyc}, as shown in I^2C Bus Timing in sec Electrical Characteristics, and as shown in table 26.10. Note that the I^2C bus interf timing specifications will not be met with a system clock frequency of less than 5

- The I^2C bus interface specification for the SCL rise time t_{sr} is under 1000 ns (300 n speed mode). In master mode, the I^2C bus interface monitors the SCL line and syn one bit at a time during communication. If t_{sr} (the time for SCL to go from low to V the time determined by the input clock of the I^2C bus interface, the high period of S extended. The SCL rise time is determined by the pull-up resistance and load capa the SCL line. To insure proper operation at the set transfer rate, adjust the pull-up and load capacitance so that the SCL rise time does not exceed the values given in below.

		mode					
1	$17.5t_{cyc}$	Standard mode	1000 ns	1000 ns	1000 ns	1000 ns	1000 n
		High-speed mode	300 ns	300 ns	300 ns	300 ns	300 ns

- The I^2C bus interface specifications for the SCL and SDA rise and fall times are unc and 300 ns. The I^2C bus interface SCL and SDA output timing is prescribed by t_{cyc}, a table 16.6. However, because of the rise and fall times, the I^2C bus interface specifi not be satisfied at the maximum transfer rate. Table 16.8 shows output timing calcu different operating frequencies, including the worst-case influence of rise and fall ti

  t_{BUFO} fails to meet the I^2C bus interface specifications at any frequency. The solution to provide coding to secure the necessary interval (approximately 1 µs) between iss stop condition and issuance of a start condition, or (b) to select devices whose input permits this output timing for use as slave devices connected to the I^2C bus.

  t_{SCLLO} in high-speed mode and t_{STASO} in standard mode fail to satisfy the I^2C bus interf specifications for worst-case calculations of t_{Sr}/t_{Sf}. Possible solutions that should be include (a) adjusting the rise and fall times by means of a pull-up resistor and capac (b) reducing the transfer rate to meet the specifications, or (c) selecting devices whc timing permits this output timing for use as slave devices connected to the I^2C bus.

		High-speed mode	–300	600	950	950	950	950
t_{SCLLO}	$0.5t_{SCLO}$ ($-t_{Sf}$)	Standard mode	–250	4700	4750	4750	4750	4750
		High-speed mode	–250	1300	1000*1	1000*1	1000*1	1000
t_{BUFO}	$0.5t_{SCLO}$ $-1t_{cyc}$ ($-t_{Sr}$)	Standard mode	–1000	4700	3800*1	3875*1	3900*1	3938
		High-speed mode	–300	1300	750*1	825*1	850*1	888*
t_{STAHO}	$0.5t_{SCLO}$ $-1t_{cyc}$ ($-t_{Sf}$)	Standard mode	–250	4000	4550	4625	4650	4688
		High-speed mode	–250	600	800	875	900	938
t_{STASO}	$1t_{SCLO}$ ($-t_{Sr}$)	Standard mode	–1000	4700	9000	9000	9000	9000
		High-speed mode	–300	600	2200	2200	2200	2200
t_{STOSO}	$0.5t_{SCLO}$ $+2t_{cyc}$ ($-t_{Sr}$)	Standard mode	–1000	4000	4400	4250	4200	4125
		High-speed mode	–300	600	1350	1200	1150	1075
t_{SDASO} (master)	$1t_{SCLLO}$*3 $-3t_{cyc}$ ($-t_{Sr}$)	Standard mode	–1000	250	3100	3325	3400	3513
		High-speed mode	–300	100	400	625	700	813
t_{SDASO} (slave)	$1t_{SCLL}$*3 $-12t_{cyc}$*2 ($-t_{Sr}$)	Standard mode	–1000	250	1300	2200	2500	2950
		High-speed mode	–300	100	–1400*1	–500*1	–200*1	250

Notes: 1. Does not meet the I^2C bus interface specification. Remedial action such as t
is necessary: (a) secure a start/stop condition issuance interval; (b) adjust th
fall times by means of a pull-up resistor and capacitive load; (c) reduce the tr
(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
CKS0 to CKS2. Depending on the frequency it may not be possible to achiev
maximum transfer rate; therefore whether or not the I^2C bus interface specifi
met must be determined in accordance with the actual setting conditions.

2. Value when the IICX bit is set to 1. When the IICX bit is cleared to 0, the valu
$-6t_{cyc}$).
3. Calculated using the I^2C bus specification values (standard mode: 4700 ns m
speed mode: 1300 ns min.).

- Note on ICDR Read at End of Master Reception

  To halt reception at the end of a receive operation in master receive mode, set the T
  and write 0 to BBSY and SCP in ICCR. This changes SDA from low to high when
  high, and generates the stop condition. After this, receive data can be read by means
  ICDR read, but if data remains in the buffer the ICDRS receive data will not be tran
  ICDR, and so it will not be possible to read the second byte of data.

  If it is necessary to read the second byte of data, issue the stop condition in master r
  mode (i.e. with the TRS bit cleared to 0). When reading the receive data, first confir
  BBSY bit in the ICCR register is cleared to 0, the stop condition has been generated
  bus has been released, then read the ICDR register with TRS cleared to 0.

  Note that if the receive data (ICDR data) is read in the interval between execution o
  instruction for issuance of the stop condition (writing of 0 to BBSY and SCP in ICC
  actual generation of the stop condition, the clock may not be output correctly in sub
  master transmission.

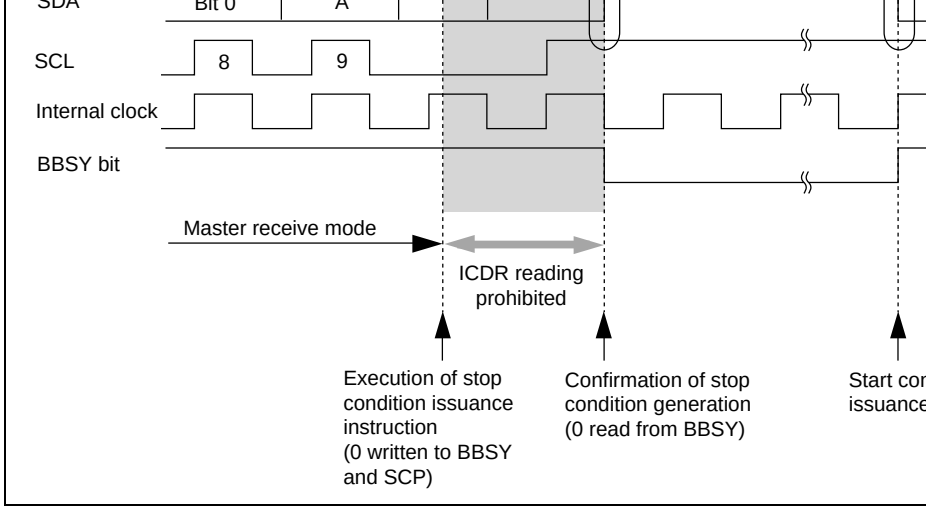
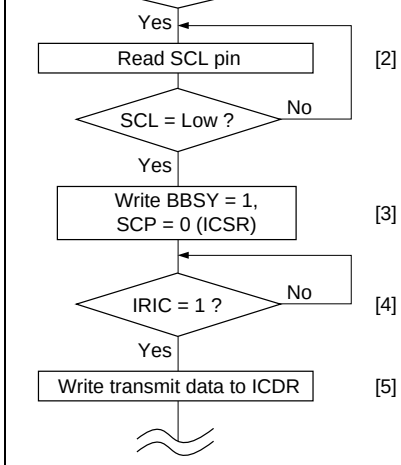


Figure 16.18 Points for Attention Concerning Reading of Master Receive

- Notes on Start Condition Issuance for Retransmission

  Figure 16.19 shows the timing of start condition issuance for retransmission, and t
  subsequently writing data to ICDR, together with the corresponding flowchart. Aft
  condition issuance is done and determined the start condition, write the transmit da
  as shown below.



[5] Set transmit data (slave address + R/W)

Note: Program so that processing from [3] to [
executed continuously.

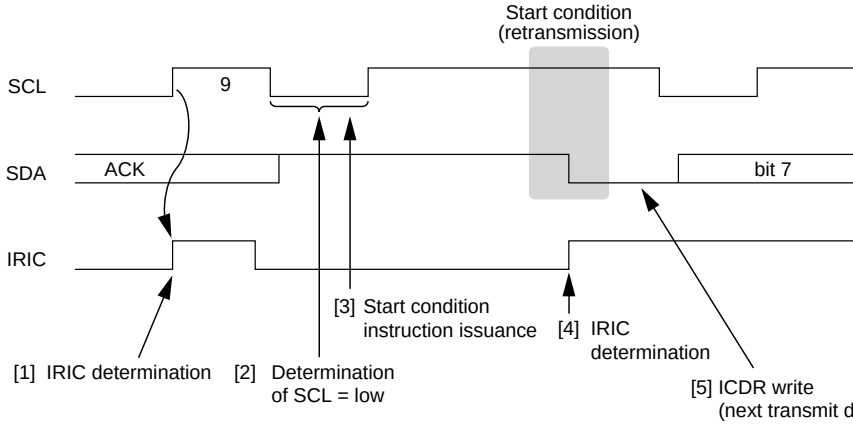


**Figure 16.19 Flowchart and Timing of Start Condition Instruction Issuanc
Retransmission**

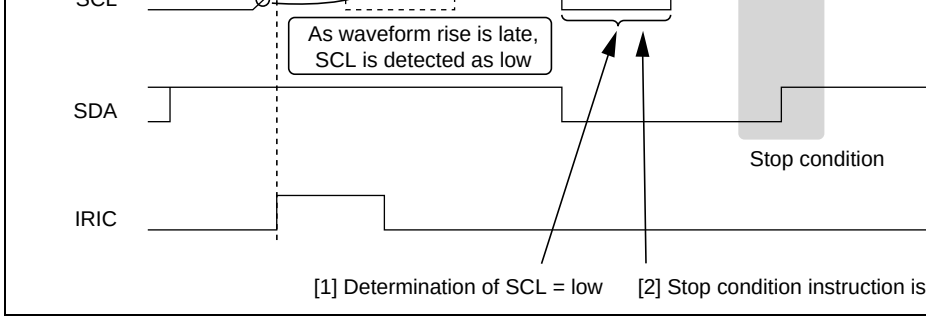


Figure 16.20 Timing of Stop Condition Issuance

— Error phenomenon

Normally, WAIT State will be cancelled by clearing the IRIC flag bit from 1 to
fall of the 8th clock in WAIT State. In this case, if the IRIC flag bit is cleared be
7th clock fall and the 8th clock fall, the IRIC flag clear- data will be retained int
Therefore, the WAIT State will be cancelled right after WAIT insertion on 8th c

— Restrictions

Please clear the IRIC flag before the rise of the 7th clock (the counter value of B
through BC0 should be 2 or greater), after the IRIC flag is set to 1 on the rise of
clock.

If the IRIC flag-clear is delayed due to the interrupt or other processes and the v
counter is turned to 1 or 0, please confirm the SCL pins are in L' state after the c
value of BC2 through BC0 is turned to 0, and clear the IRIC flag. (See figure 16

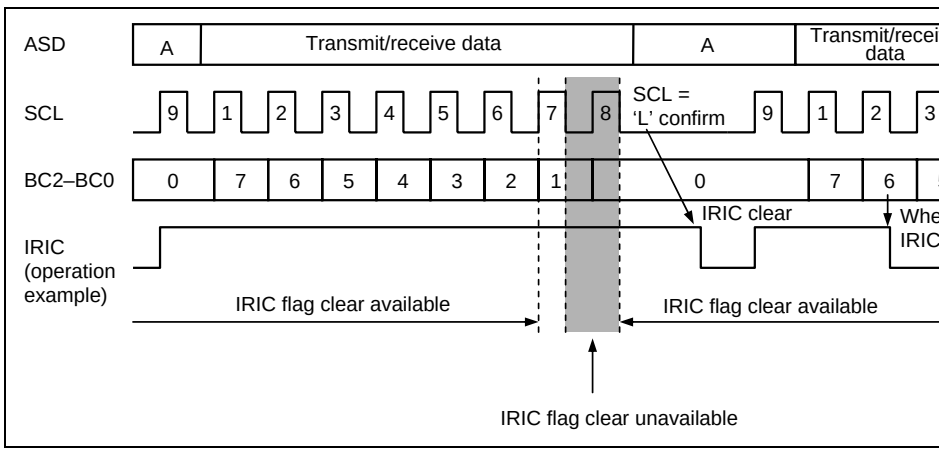


Figure 16.21 IRIC Flag Clear Timing on WAIT Operation

To ensure that the interrupt processing is performed properly, one of the following conditions should be applied.

(1) Make sure that reading received data from the ICDR register, or reading or wri ICCR register, is completed before the next slave address receive operation sta

(2) Monitor the BC2 to BC0 counter in the ICMR register and, when the value of I is 000 (8th or 9th clock cycle), allow a waiting time of at least 2 transfer clock order to involve the problem period in question before reading from the ICDR reading or writing to the ICCR register.

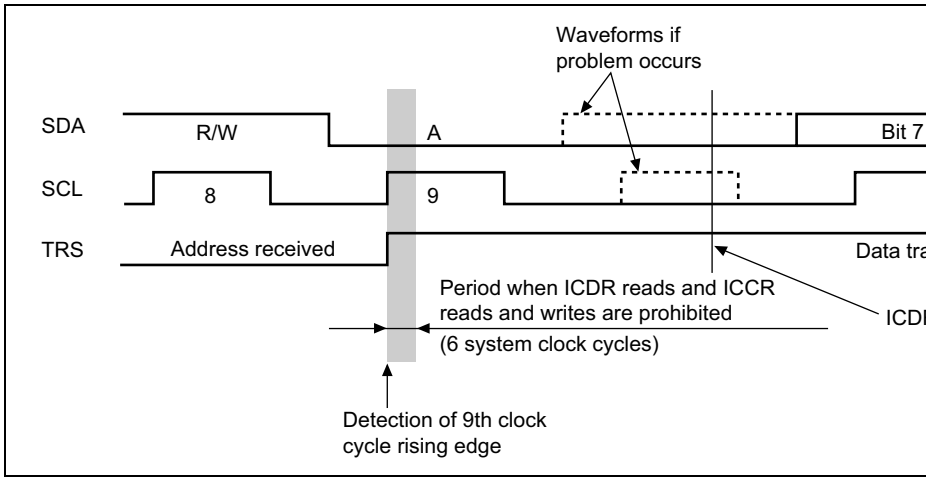


Figure 16.22 ICDR Read and ICCR Access Timing in Slave Transmit M

This results in the actual internal value of the TRS bit remaining 1 (transmit mode) acknowledge bit being sent at the 9th clock cycle address receive completion in the address receive operation following a restart condition input with no stop condition intervening.

When receiving an address in the slave mode, clear the TRS bit to 0 during the peri indicated as (a) in figure 16.23.

To cancel the holding of the SCL bit low by the wait function in the slave mode, cle bit to 0 and then perform a dummy read of the ICDR register.

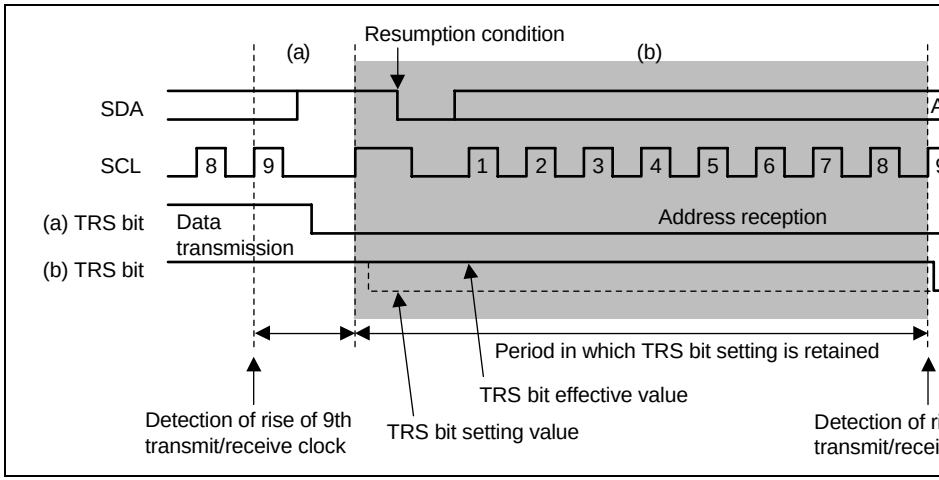


Figure 16.23 TRS Bit Setting Timing in Slave Mode

figure 16.24.)

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 on data has been transmitted or received.

When arbitration is lost during transmitting the second frame or subsequent frame, avoidance measures.

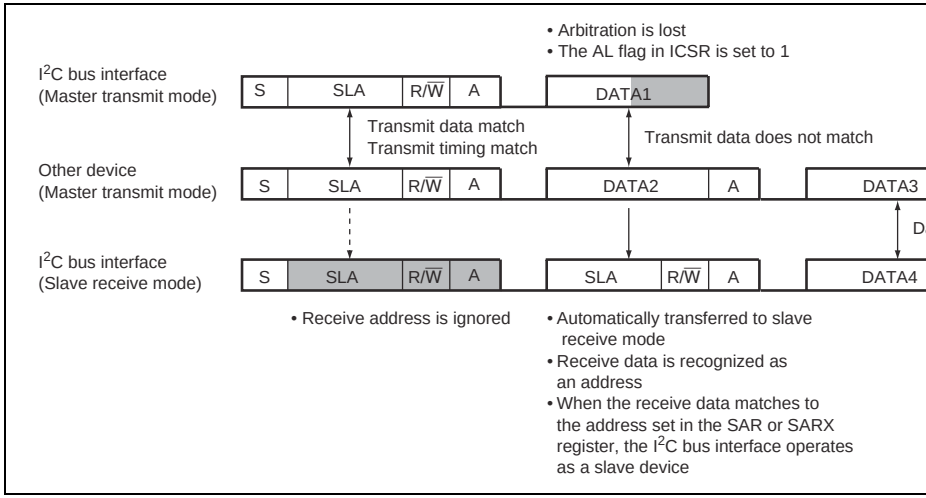


Figure 16.24 Diagram of Erroneous Operation when Arbitration is Lo

Though it is prohibited in the normal I²C protocol, the same problem may occur w bit is erroneously set to 1 and a transition to master mode is occurred during data t or reception in slave mode. In multi-master mode, pay attention to the setting of th when a bus conflict may occur. In this case, the MST bit in the ICCR register shou according to the order below.

(a) Make sure that the BBSY flag in the ICCR register is 0 and the bus is free befo the MST bit.

(b) Set the MST bit to 1.

to 1

- Rising edge of the 9th transmit/receive clock is input to the SCL pin

When the above two conditions are satisfied in slave receive mode, an unnecess interrupt occurs.

Figure 16.25 shows the note on interrupt occurrence in slave mode after receivi acknowledge bit (ACKB = 1).

(1) For the last transmit data in master transmit mode or slave transmit mode, 1 as the acknowledge bit.

If the ACKE bit in ICCR is set to 1 at this time, the ACKB bit in ICSR is set

(2) After switching to slave receive mode, the start condition is input, and addre reception is performed next.

(3) Even if the received address does not match the address set in SAR or SARX flag is set to 1 at the rise of the 9th transmit/receive clock, thus causing an in occur.

Note that if the slave address matches, an interrupt is to be generated at the rise transmit/receive clock as normal operation, so this is not erroneous operation.

— Restriction

In a transmit operation of the I^2C bus interface module, carry out the following countermeasures.

(1) After 1 is received as the acknowledge bit for transmit data, clear the ACKE ICCR to 0 to clear the ACKB bit to 0.

(2) To enable acknowledge bit reception afterwards, set the ACKE bit to 1 again

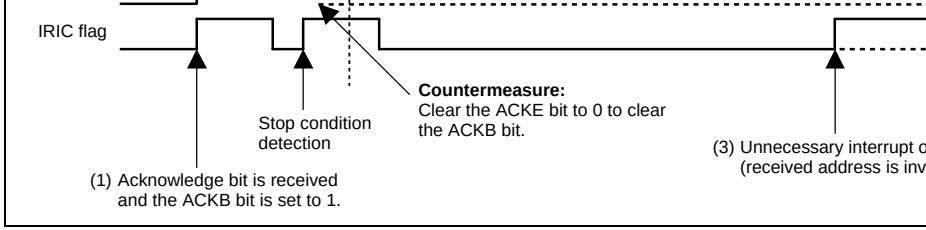


Figure 16.25 Note on Interrupt Occurrence in Slave Mode after ACKB = 1 R

(1) When previously received 2-bytes data remains in ICDR unread (ICDRS are

(2) Reads ICDR register after switching to transmit mode (TRS = 1). (RDRF = 0

(3) Sets to receive mode (TRS = 0), after transmitting Rev.1 frame of issued sta by master mode.

— Slave mode

Figure 16.27 shows the notes on ICDR writing (TRS = 0) in slave mode.

(1) Writes ICDR register in receive mode (TRS = 0), after entering the start con slave mode (TDRE = 0 state).

Address match with Rev.1 frame, receive 1 by R/W bit, and switches to tran (TRS = 1).

When these conditions are satisfied, the low fixation of the SCL pins is canc without ICDR register access after Rev.1 frame is transferred.

— Restriction

Please carry out the following countermeasures when transmitting/receiving via interface module.

(1) Please read the ICDR registers in receive mode, and write them in transmit m

(2) In receiving operation with master mode, please issue the start condition afte the internal flag of the IIC bus interface module, using CLR3 to CLR0 bit of DDCSWR register on bus-free state (BBSY = 0).

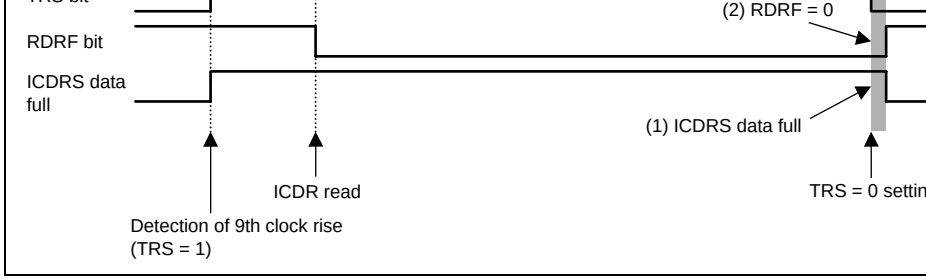


Figure 16.26 Notes on ICDR Reading with TRS = 1 Setting in Master M

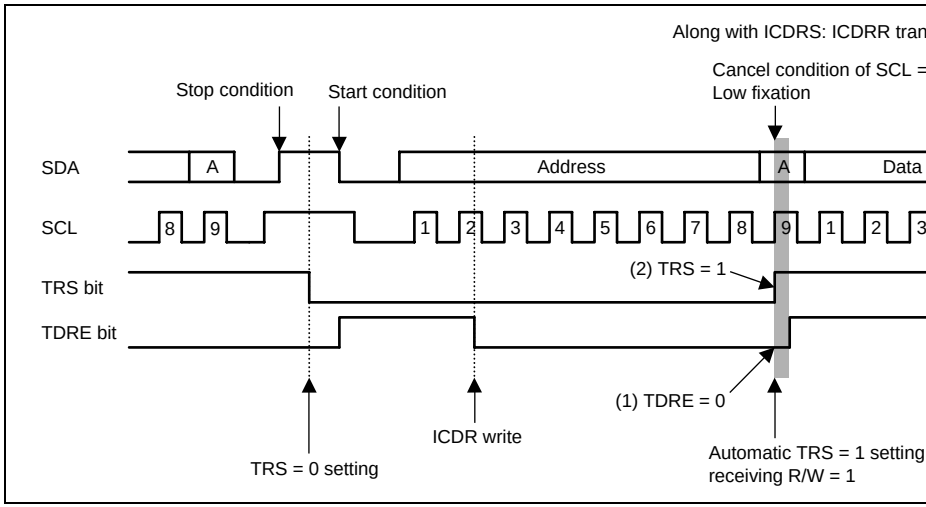


Figure 16.27 Notes on ICDR Writing with TRS = 0 Setting in Slave Mo

designated 0, 1, and 2. The keyboard buffer controller is provided with functions com
the PS/2 interface specifications.

Data transfer using the keyboard buffer controller employs a data line (KD) and a cloc
providing economical use of connectors, board surface area, etc. Figure 17.1 shows ho
keyboard buffer controller is connected.

17.1.1 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

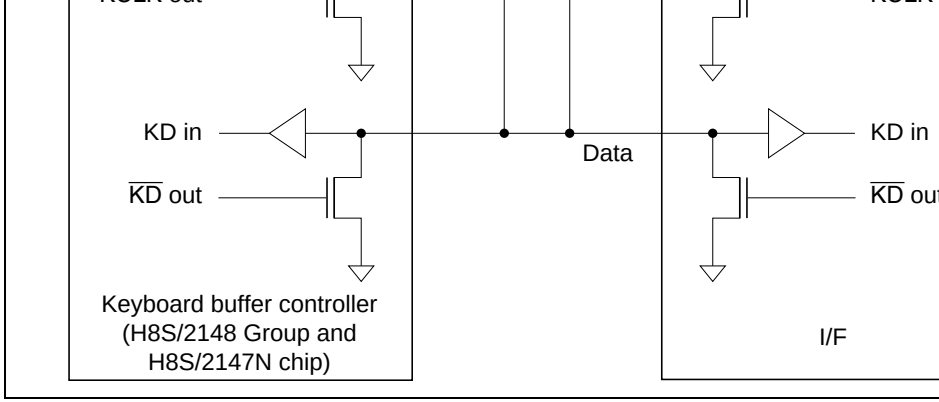
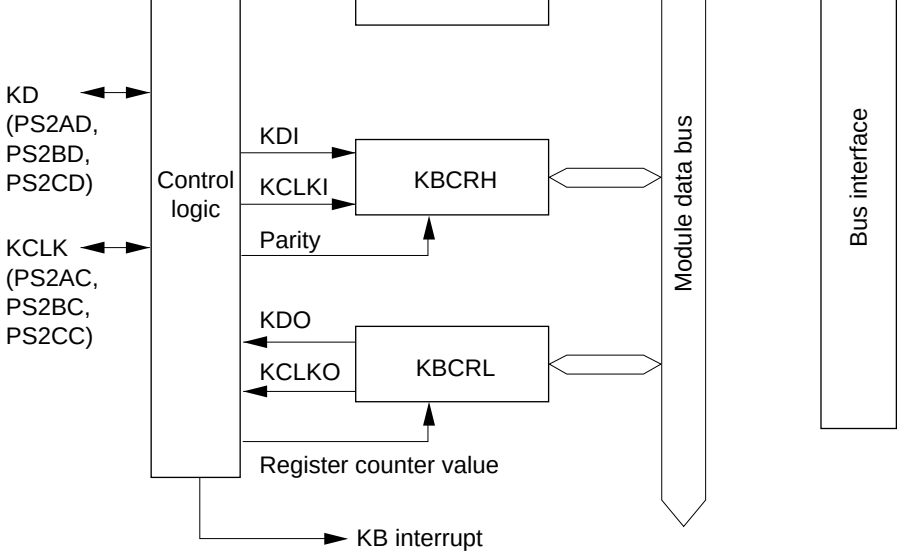


Figure 17.1 Keyboard Buffer Controller Connection



Legend:
KD: KBC data I/O pin
KCLK: KBC clock I/O pin
KBBR: Keyboard data buffer register
KBCRH: Keyboard control register H
KBCRL: Keyboard control register L

Figure 17.2 Block Diagram of Keyboard Buffer Controller

1	KBC clock I/O pin (KCLK1)	PS2BC	I/O	KBC clock inp
	KBC data I/O pin (KD1)	PS2BD	I/O	KBC data inp
2	KBC clock I/O pin (KCLK2)	PS2CC	I/O	KBC clock inp
	KBC data I/O pin (KD2)	PS2CD	I/O	KBC data inp

Note: * These are the external I/O pin names. In the text, clock I/O pins are referred and data I/O pins as KD, omitting the channel designations.

17.1.4 Register Configuration

Table 17.2 lists the registers of the keyboard buffer controller.

Table 17.2 Keyboard Buffer Controller Registers

Channel	Name	Abbreviation	R/W	Initial Value
0	Keyboard control register H	KBCRH0	R/(W)*2	H'70
	Keyboard control register L	KBCRL0	R/W	H'70
	Keyboard data buffer register	KBBR0	R	H'00
1	Keyboard control register H	KBCRH1	R/(W)*2	H'70
	Keyboard control register L	KBCRL1	R/W	H'70
	Keyboard data buffer register	KBBR1	R	H'00
2	Keyboard control register H	KBCRH2	R/(W)*2	H'70
	Keyboard control register L	KBCRL2	R/W	H'70
	Keyboard data buffer register	KBBR2	R	H'00
Common	Module stop control register	MSTPCRH	R/W	H'3F
		MSTPCRL	R/W	H'FF

Notes: 1. Lower 16 bits of the address.
2. Only 0 can be written in bits 2 and 1, to clear the flags.

Read/Write	R/W	R	R	R/W	R/W	R/(W)*	R/(W)*

Note: * Only 0 can be written, to clear the flags.

KBCRH is an 8-bit readable/writable register that indicates the operating status of the buffer controller.

KBCRH is initialized to H'70 by a reset, and in standby mode, watch mode, subactive subsleep mode, and module stop mode. Bits 6, 5, and 2 to 0 are also initialized when I cleared to 0.

Bit 7—Keyboard In/Out Enable (KBIOE): Selects whether or not the keyboard buf controller is used. When KBIOE is set to 1, the keyboard buffer controller is enabled f transmission and reception and the port pins function as KCLK and KD I/O pins. Whe cleared to 0, the keyboard buffer controller stops functioning and the port pins go to th impedance state.

Bit 7

KBIOE	Description
0	The keyboard buffer controller is non-operational (KCLK and KD signal port functions) (
1	The keyboard buffer controller is enabled for transmission and reception KD signal pins are in the bus drive state)

Bit 6—Keyboard Clock In (KCLKI): Monitors the KCLK I/O pin. This bit cannot b

Bit 6

KCLKI	Description
0	KCLK I/O pin is low
1	KCLK I/O pin is high (

Bit 4—Keyboard Buffer Register Full Select (KBFSEL): 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 KBE bit in the KBCRL register should be cleared to 0 to disable recep

Bit 4	
KBFSEL	**Description**
0	KBF bit is used as KCLK fall interrupt flag
1	KBF bit is used as keyboard buffer register full flag (In

Bit 3—Keyboard Interrupt Enable (KBIE): Enables or disables interrupts from the k buffer controller to the CPU.

Bit 3	
KBIE	**Description**
0	Interrupt requests are disabled (In
1	Interrupt requests are enabled

Bit 2—Keyboard Buffer Register Full (KBF): Indicates that data reception has been and the received data is in the keyboard data buffer register (KBBR).

Bit 2	
KBF	**Description**
0	[Clearing condition] (In Read KBF when KBF =1, then write 0 in KBF
1	[Setting conditions] • When data has been received normally and has been transferred to K (keyboard buffer register full flag) • When a KCLK falling edge is detected (while KBFSEL = 0) (KCLK int

Bit 0—Keyboard Stop (KBS): Indicates the receive data stop bit. Valid only when K

Bit 0

KBS	Description
0	0 stop bit received
1	1 stop bit received

17.2.2 Keyboard Control Register L (KBCRL)

Bit	7	6	5	4	3	2	1
	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1
Initial value	0	1	1	1	0	0	0
Read/Write	R/W	R/W	R/W	—	R	R	R

KBCRL is an 8-bit readable/writable register that enables the receive counter count an the keyboard buffer controller pin output.

KBCRL is initialized to H'70 by a reset, and in standby mode, watch mode, subactive subsleep mode, and module stop mode.

Bit 7—Keyboard Enable (KBE): Enables or disables loading of receive data into the data buffer register (KBBR).

Bit 7

KBE	Description
0	Loading of receive data into KBBR is disabled
1	Loading of receive data into KBBR is enabled

Bit 5	
KDO	Description
0	Keyboard buffer controller data I/O pin is low
1	Keyboard buffer controller data I/O pin is high (In

Bit 4—Reserved: This bit cannot be modified and is always read as 1.

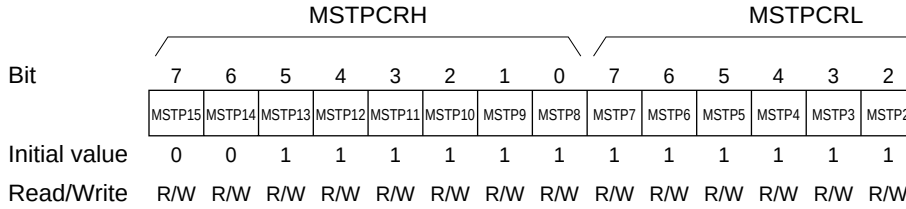
Bits 3 to 0—Receive Counter (RXCR3 to RXCR0): These bits indicate the received
Their value is incremented on the fall of KCLK. These bits cannot be modified.

The receive counter is initialized to 0000 by a reset and when 0 is written in KBE. Its v
to 0000 after a stop bit is received.

Bit 3	Bit 2	Bit 1	Bit 0		
RXCR3	RXCR2	RXCR1	RXCR0	Receive Data Contents	
0	0	0	0	—	(In
			1	Start bit	
		1	0	KB0	
			1	KB1	
	1	0	0	KB2	
			1	KB3	
		1	0	KB4	
			1	KB5	
1	0	0	0	KB6	
			1	KB7	
		1	0	Parity bit	
			1	—	
	1	—	—	—	

KBBR is initialized to H'00 by a reset, in standby mode, watch mode, subactive mode mode, and module stop mode, and when KBIOE is cleared to 0.

17.2.4 Module Stop Control Register (MSTPCR)

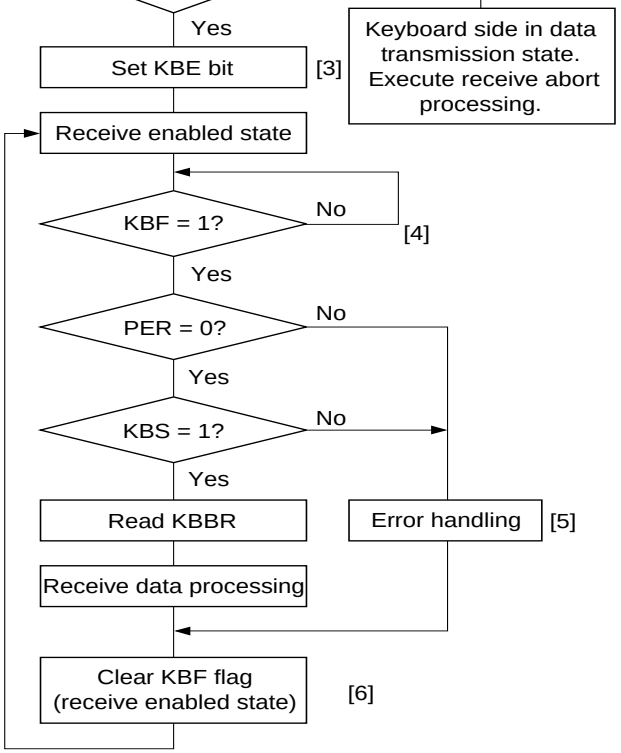


MSTPCR, comprising two 8-bit readable/writable register, performs module stop mod When the MSTP2 bit is set to 1, the keyboard buffer controller halts and enters modul See section 25.5, Module Stop Mode, for details.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not in software standby mode.

MSTPCRL Bit 2—Module Stop (MSTP2): Specifies keyboard buffer controller mo mode.

MSTPCRL Bit 2	
MSTP2	**Description**
0	Keyboard buffer controller module stop mode is cleared
1	Keyboard buffer controller module stop mode is set (



[3] Detect the start bit on the keyboard s receive data in synchronization w of KCLK.

[4] When a stop bit is the keyboard buffe controller drives K to disable keyboar transmission (auto inhibit).
If the KBIE bit is s KBCRH, an interru request is sent to t at the same time.

[5] Perform receive da processing.

[6] Clear the KBF flag KBCRL. At the sa the system automa drives KCLK high, the receive enable

The receive operation continued by repeatin [3] to [6].

Figure 17.3 Sample Receive Processing Flowchart

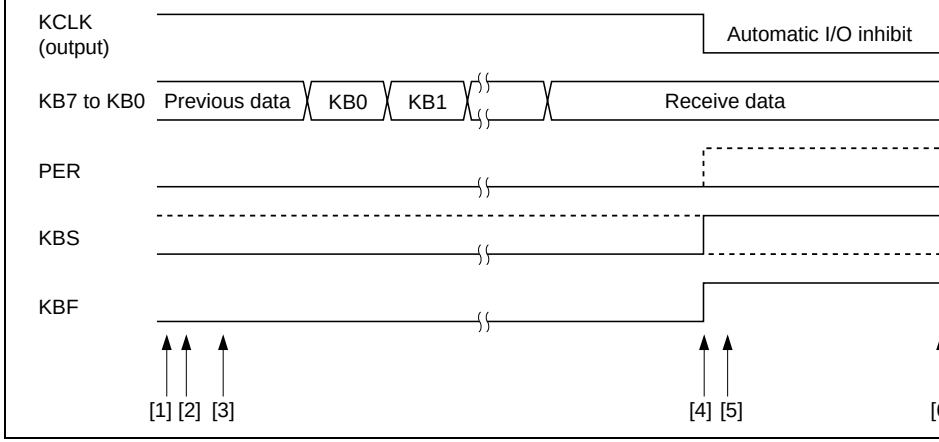
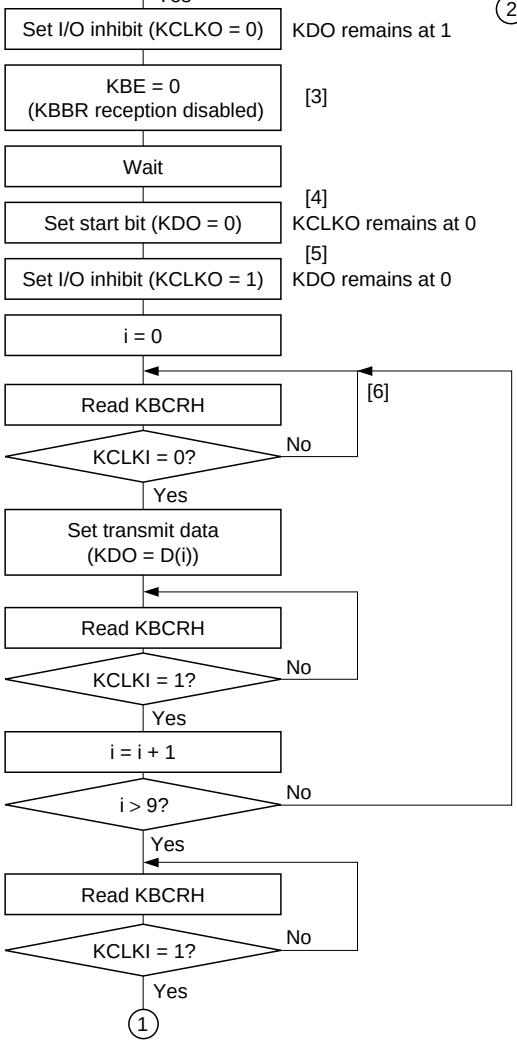


Figure 17.4 Receive Timing

17.3.2 Transmit Operation

In a transmit operation, KCLK (clock) is an output on the keyboard side, and KD (data) output on the H8S/2148 Group and H8S/2147N chip (system) side. KD outputs a start b bits (LSB-first), an odd parity bit, and a stop bit, in that order. The KD value is valid wh is high. A sample transmit processing flowchart is shown in figure 17.5, and the transm figure 17.6.



[4] Write 0 in the KDO bit (set sta

[5] Write 1 in the KCLKO bit (clea inhibit).

[6] Read KBCRH, and when KCL set the transmit data in the KD (LSB-first). Next, set the parity stop bit in the KDO bit.

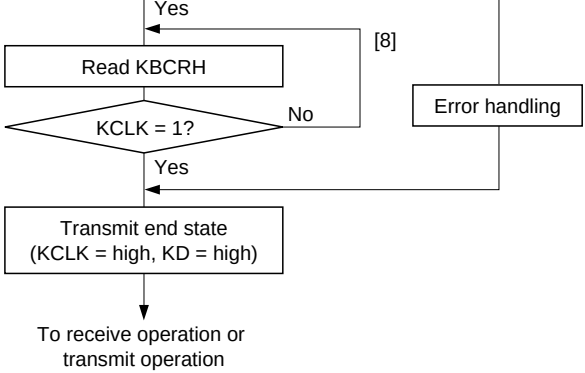
[7] After transmitting the stop bit, KBCRL and confirm that KDI (receive completed notificatio keyboard).

[8] Read KBCRH. Confirm that t and KDI bits are both 1.

The transmit operation can be co by repeating steps [2] to [8].

Notes: i = 0 to 7: Transmit data
i = 8: Parity bit
i = 9: Stop bit

Figure 17.5 Sample Transmit Processing Flowchart



Note: * To switch to reception after transmission, set KBE to 1 (KBBR receive enable) while KCLK

Figure 17.5 Sample Transmit Processing Flowchart (cont)

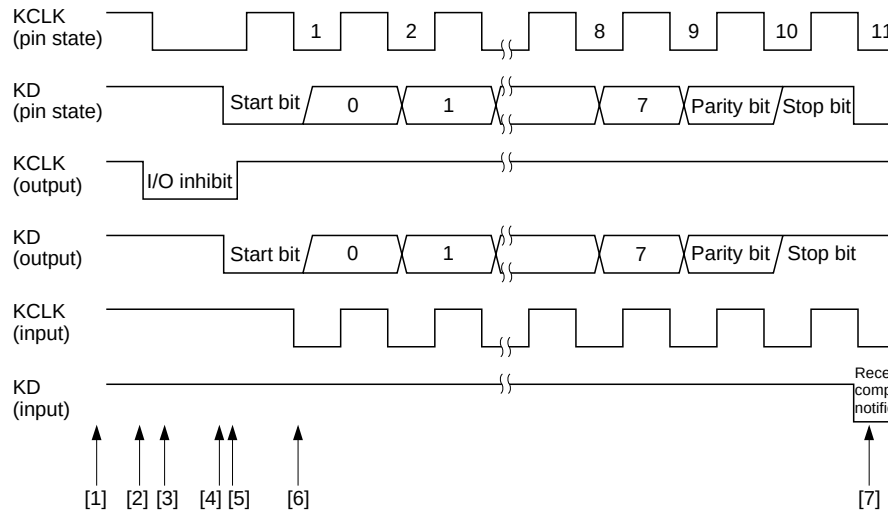


Figure 17.6 Transmit Timing

clock low for a certain period. A sample receive abort processing flowchart is shown i
17.7, and the receive abort timing in figure 17.8.

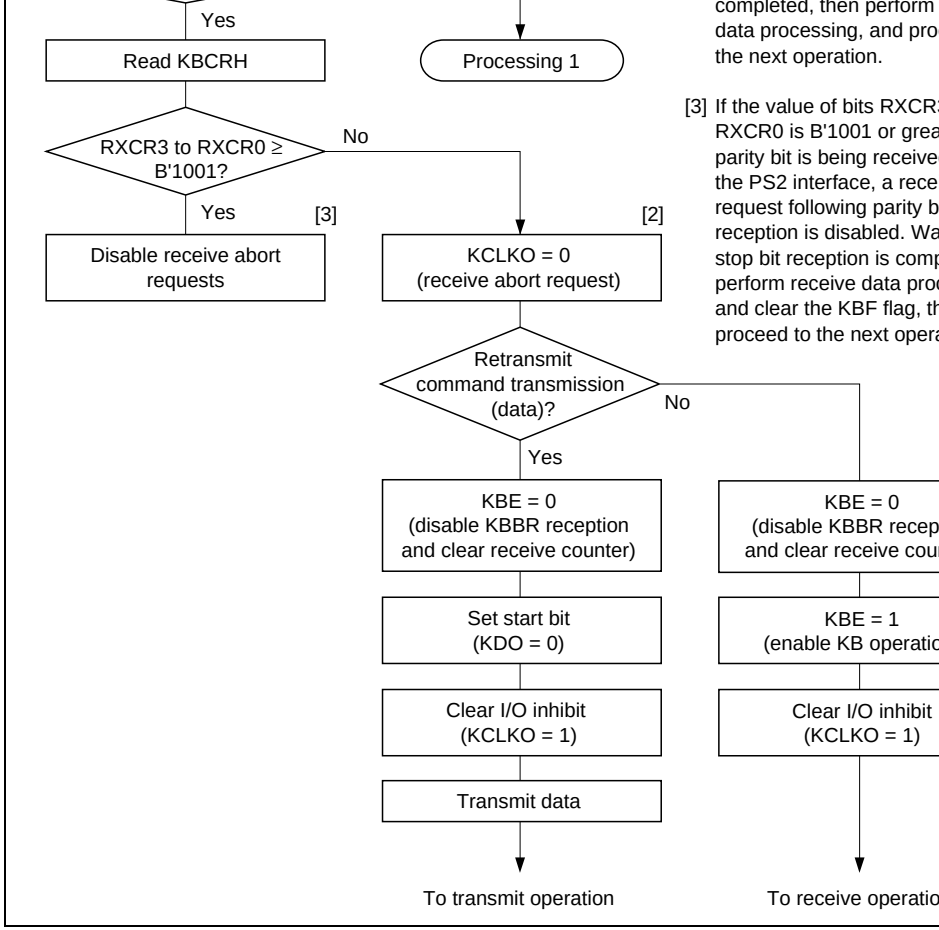
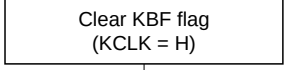


Figure 17.7 Sample Receive Abort Processing Flowchart



Transmit enabled state.
If there is transmit data,
the data is transmitted.

Figure 17.7 Sample Receive Abort Processing Flowchart (cont)

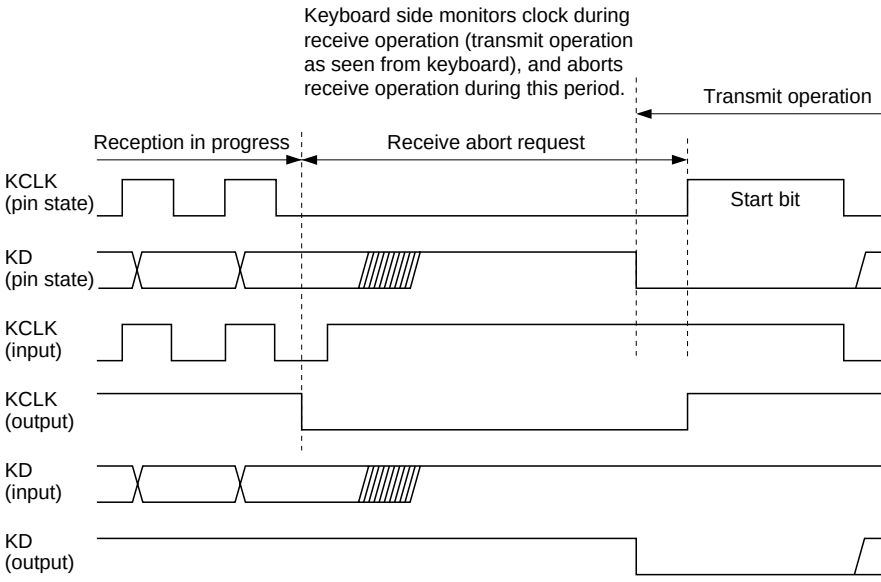
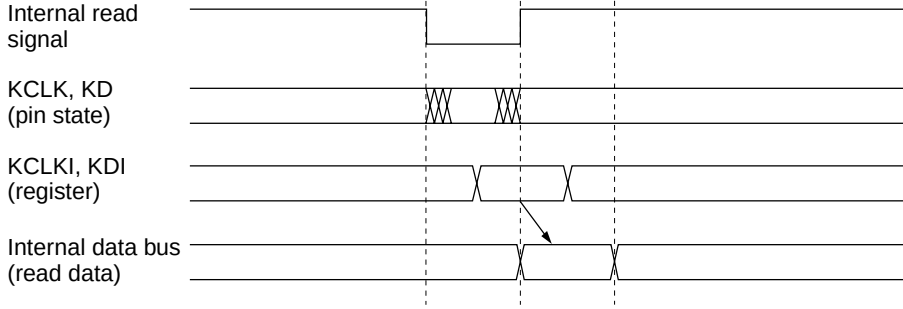
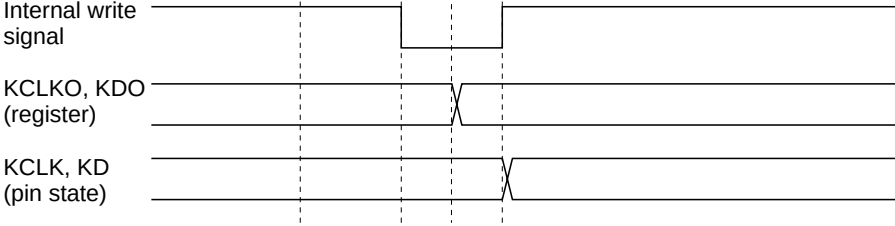


Figure 17.8 Receive Abort and Transmit Start (Transmission/Reception Swi Timing



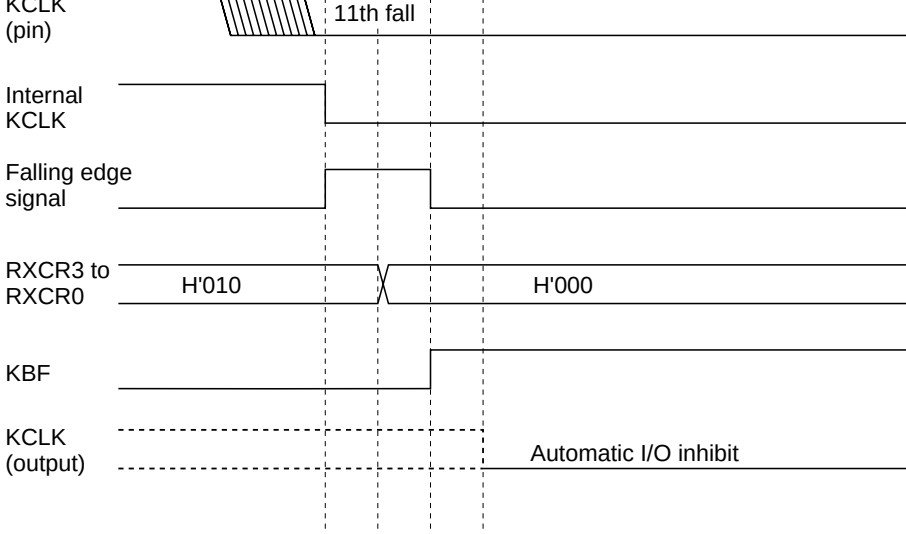
Note: * The ϕ clock shown here is scaled by 1/N in medium-speed mode when the ope mode is active mode.

Figure 17.9 KCLKI and KDI Read Timing



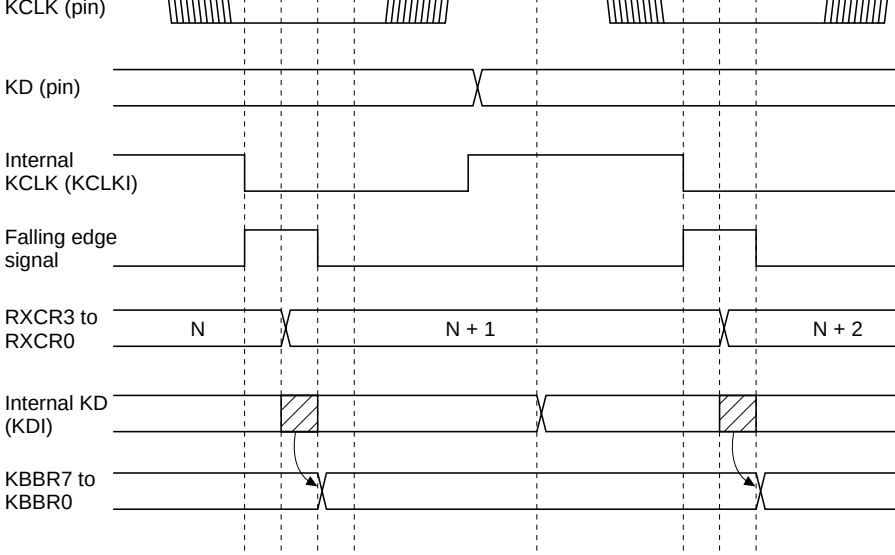
Note: * The φ clock shown here is scaled by 1/N in medium-speed mode when the op
mode is active mode.

Figure 17.10 KCLKO and KDO Write Timing



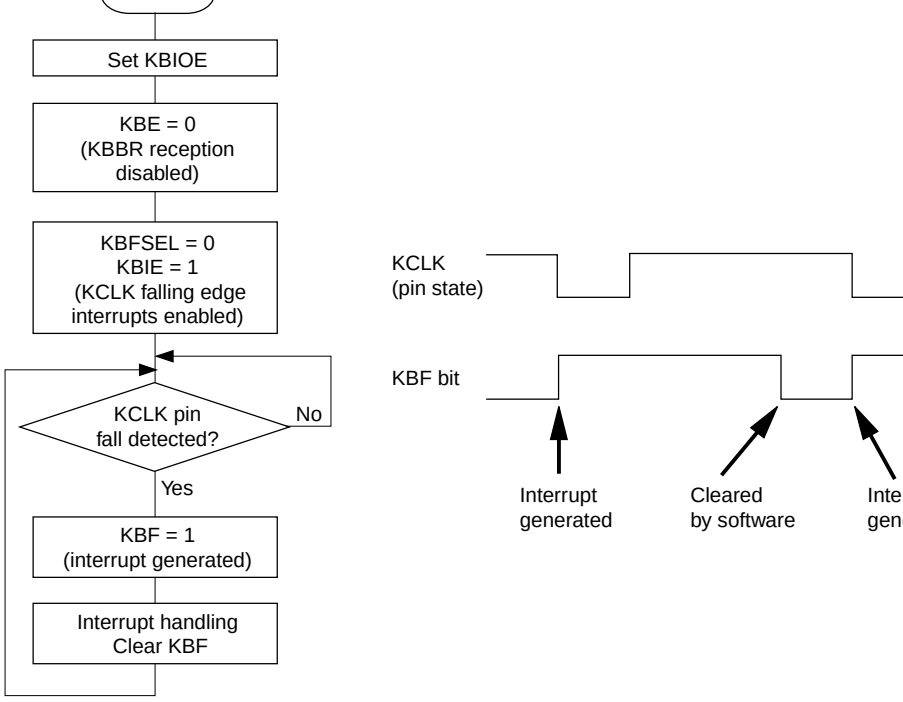
Note: * The ϕ clock shown here is scaled by 1/N in medium-speed mode when the ope
mode is active mode.

Figure 17.11 KBF Setting and KCLK Automatic I/O Inhibit Generation Ti



Note: * The ϕ clock shown here is scaled by 1/N in medium-speed mode when the operating mo
mode.

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 inh
generation in figure 17.11. When the KBF bit is used as the KCLK input fall interrupt flag, the
automatic I/O inhibit function does not operate.

Figure 17.13 Example of KCLK Input Fall Interrupt Operation

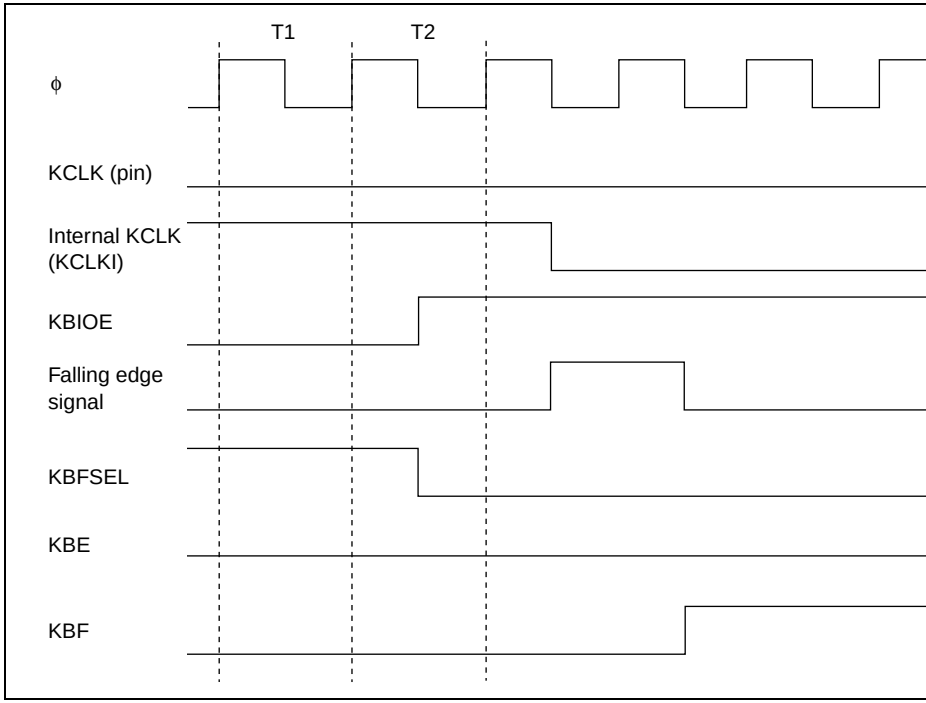


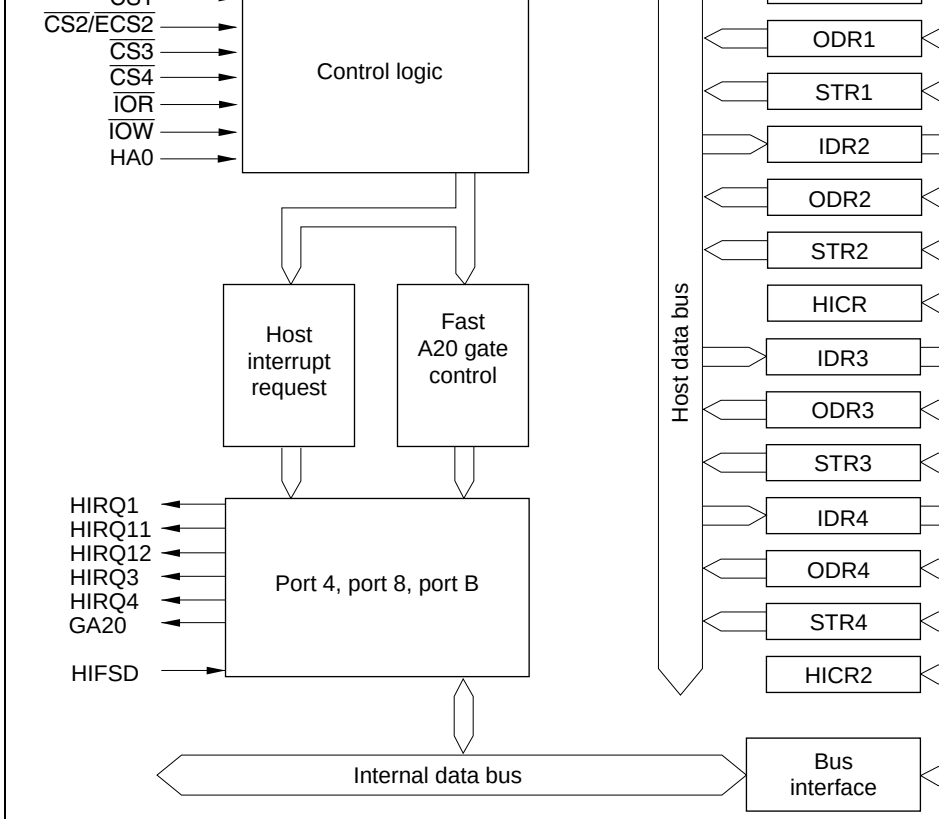
Figure 17.14 KBIOE Setting and KCLK Falling Edge Detection Timi

connection to an ISA bus, widely used as the internal bus in personal computers. The
interface provides a four-channel parallel interface between the on-chip CPU and a ho
The host interface is available only when the HI12E bit is set to 1 in SYSCR2. This m
slave mode, because it is designed for a master-slave communication system in which
H8S/2148 Group and H8S/2147N chip is slaved to a host processor.

18.1.1 Features

The features of the host interface are summarized below.

The host interface consists of 8-byte data registers, 4-byte status registers, a 2-byte co
register, fast A20 gate logic, and a host interrupt request circuit. Communication is ca
seven control signals from the host processor ($\overline{CS1}$, $\overline{CS2}$ or $\overline{ECS2}$, $\overline{CS3}$, $\overline{CS4}$, HA0, $\overline{IO}$
$\overline{IOW}$), six output signals to the host processor (GA20, HIRQ1, HIRQ11, HIRQ12, HI
HIRQ4), and an 8-bit bidirectional command/data bus (HDB7 to HDB0). The $\overline{CS1}$, $\overline{CS}$
$\overline{ECS2}$), $\overline{CS3}$, and $\overline{CS4}$ signals select one of the four interface channels.



Legend:

IDR1:	Input data register 1	IDR3:	Input data register 3
IDR2:	Input data register 2	IDR4:	Input data register 4
ODR1:	Output data register 1	ODR3:	Output data register 3
ODR2:	Output data register 2	ODR4:	Output data register 4
STR1:	Status register 1	STR3:	Status register 3
STR2:	Status register 2	STR4:	Status register 4
HICR:	Host interface control register 1	HICR2:	Host interface control register 2

Figure 18.1 Block Diagram of Host Interface

Chip select 1	$\overline{CS1}$	P95	Input	Host interface chip select signal ODR1, STR1
Chip select 2*	$\overline{CS2}$	P81	Input	Host interface chip select signal ODR2, STR2
	$\overline{ECS2}$	P90		
Chip select 3	$\overline{CS3}$	PB2	Input	Host interface chip select signal ODR3, STR3
Chip select 4	$\overline{CS4}$	PB3	Input	Host interface chip select signal ODR4, STR4
Command/data	HA0	P80	Input	Host interface address select sig In host read access, this signal s status registers (STR1 to STR4) registers (ODR1 to ODR4). In ho access to the data registers (IDR and IDTR4), this signal indicates the host is writing a command o
Data bus	HDB7 to HDB0	P37 to P30	I/O	Host interface data bus
Host interrupt 1	HIRQ1	P44	Output	Interrupt output 1 to host
Host interrupt 11	HIRQ11	P43	Output	Interrupt output 11 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

Note: * Selection of $\overline{CS2}$ or $\overline{ECS2}$ is by means of the CS2E bit in STCR and the FG HICR. Host interface channel 2 and the $\overline{CS2}$ pin can be used when CS2E = CS2E = 1, $\overline{CS2}$ is used when FGA20E =0, and $\overline{ECS2}$ is used when FGA20 manual, both are referred to as $\overline{CS2}$.

System control register	SYSCR	R/W*1	—	H'09	H'FFC4	—	—	—
System control register 2	SYSCR2	R/W	—	H'00	H'FF83	—	—	—
Host interface control register 1	HICR	R/W	—	H'F8	H'FFF0	—	—	—
Host interface control register 2	HICR2	R/W	—	H'F8	H'FE80	—	—	—
Input data register 1	IDR1	R	W	—	H'FFF4	0	1	1
Output data register 1	ODR1	R/W	R	—	H'FFF5	0	1	1
Status register 1	STR1	R/(W)*2	R	H'00	H'FFF6	0	1	1
Input data register 2	IDR2	R	W	—	H'FFFC	1	0	1
Output data register 2	ODR2	R/W	R	—	H'FFFD	1	0	1
Status register 2	STR2	R/(W)*2	R	H'00	H'FFFE	1	0	1
Input data register 3	IDR3	R	W	—	H'FE84	1	1	0
Output data register 3	ODR3	R/W	R	—	H'FE85	1	1	0
Status register 3	STR3	R/(W)*2	R	H'00	H'FE86	1	1	0
Input data register 4	IDR4	R	W	—	H'FE8C	1	1	1
Output data register 4	ODR4	R/W	R	—	H'FE8D	1	1	1
Status register 4	STR4	R/(W)*2	R	H'00	H'FE8E	1	1	1
Module stop control register	MSTPCRH	R/W	—	H'3F	H'FF86	—	—	—
	MSTPCRL	R/W	—	H'FF	H'FF87	—	—	—

Notes: 1. Bits 5 and 3 are read-only bits.

2. The user-defined bits (bits 7 to 4 and 2) are read/write accessible from the s processor.

3. Address when accessed from the slave processor. The lower 16 bits of the a shown.

4. Pin inputs used in access from the host processor.

5. The HA0 input discriminates between writing of commands and data.

Read/Write	R/W	R/W	R	R/W	R	R/W	R/W

SYSCR is an 8-bit readable/writable register which controls H8S/2148 Group chip op
the host interface registers, HICR, IDR1, ODR1, STR1, IDR2, ODR2, and STR2 can
accessed when the HIE bit is set to 1. HICR2, IDR3, ODR3, STR3, IDR4, ODR4, and
be accessed regardless of the setting of the HIE bit. The host interface $\overline{CS2}$ and $\overline{ECS2}$
controlled by the CS2E bit in SYSCR and the FGA20E bit in HICR. See section 3.2.2
Control Register (SYSCR), and section 5.2.1, System Control Register (SYSCR), for
on other SYSCR bits. SYSCR is initialized to H'09 by a reset and in hardware standby

Bit 7—CS2 Enable Bit (CS2E): Used together with the FGA20E bit in HICR to sele
that performs the $\overline{CS2}$ function.

SYSCR Bit 7	HICR Bit 0	
CS2E	**FGA20E**	**Description**
0	0	$\overline{CS2}$ pin function halted ($\overline{CS2}$ fixed high internally)
	1	
1	0	$\overline{CS2}$ pin function selected for P81/$\overline{CS2}$ pin
	1	$\overline{CS2}$ pin function selected for P90/$\overline{ECS2}$ pin

1	Host interface register (HICR, IDR1, ODR1, STR1, IDR2, ODR2, STR2), C access is enabled

18.2.2 System Control Register 2 (SYSCR2)

Bit	7	6	5	4	3	2	1
	KWUL1	KWUL0	P6PUE	—	SDE	CS4E	CS3E
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	—	R/W	R/W	R/W

SYSCR2 is an 8-bit readable/writable register which controls chip operations. Host inte functions are enabled or disabled by the HI12E bit in SYSCR2. The number of channel be used can be extended to a maximum of four by means of the CS3E bit and CS4E bit is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 and 6—Key Wakeup Level 1 and 0 (KWUL1, KWUL0): The port 6 input lev set and changed by software. For details see section 8, I/O Ports.

Bit 5—Port 6 Input Pull-Up Extra (P6PUE): Controls and selects the current specifi the port 6 MOS input pull-up function connected by means of KMPCR settings. For de section 8, I/O Ports.

Bit 4—Reserved: Do not write 1 to this bit.

Bit 2—CS4 Enable (CS4E): Enables or disables host interface channel 4 functions in
When these functions are enabled, channel 4 pins are enabled and processing can be p
data transfer between the slave and the host.

Bit 2	
CS4E	**Description**
0	Host interface pin channel 4 functions disabled
1	Host interface pin channel 4 functions enabled

Bit 1—CS3 Enable (CS3E): Enables or disables host interface channel 3 functions in
When these functions are enabled, channel 3 pins are enabled and processing can be p
data transfer between the slave and the host.

Bit 1	
CS3E	**Description**
0	Host interface pin channel 3 functions disabled
1	Host interface pin channel 3 functions enabled

Bit 0—Host Interface Enable Bit (HI12E): Enables or disables host interface functi
single-chip mode. When the host interface functions are enabled, slave mode is entere
processing is performed for data transfer between the slave and host.

Bit 0	
HI12E	**Description**
0	Host interface functions are disabled
1	Host interface functions are enabled

Host Read/Write	—	—	—	—	—	—	—

- HICR2

Bit	7	6	5	4	3	2	1
	—	—	—	—	—	IBFIE4	IBFIE3
Initial value	1	1	1	1	1	0	0
Slave Read/Write	—	—	—	—	—	R/W	R/W
Host Read/Write	—	—	—	—	—	—	—

HICR is an 8-bit readable/writable register which controls host interface channel 1 and and the fast A20 gate function. HICR2 is an 8-bit readable/writable register which cont interface channel 3 and 4 interrupts. HICR and HICR2 are initialized to H'F8 by a reset hardware standby mode.

Bits 7 to 3—Reserved: These bits cannot be modified and are always read as 1.

HICR Bits 2 and 1—Input Data Register Full Interrupt Enable 2 and 1 (IBFIE2, I
HICR2 Bits 2 and 1—Input Data Register Full Interrupt Enable 4 and 3 (IBFIE4,
These bits enable or disable the IBF1, IBF2, IBF3, and IBF4 interrupts to the internal C

				request disabled (
—	—	1	—	Input data register (IDR2) reception completed request enabled
—	0	—	—	Input data register (IDR3) reception completed request disabled (
—	1	—	—	Input data register (IDR3) reception completed request enabled
0	—	—	—	Input data register (IDR4) reception completed request disabled (
1	—	—	—	Input data register (IDR4) reception completed request enabled

HICR Bit 0—Fast A20 Gate Function Enable (FGA20E): Enables or disables the f function. When the fast A20 gate is disabled, the normal A20 gate can be implemente firmware operation of the P81 output.

HICR Bit 0	
FGA20E	**Description**
0	Fast A20 gate function disabled (I
1	Fast A20 gate function enabled

HICR2 Bit 0—Reserved: Do not set this bit to 1.

IDR1 is an 8-bit read-only register to the slave processor, and an 8-bit write-only regist
host processor. When $\overline{CSn}$ (n = 1 to 4) is low, information on the host data bus is writte
IDRn at the rising edge of $\overline{IOW}$. The HA0 state is also latched into the C/$\overline{D}$ bit in STRn
whether the written information is a command or data.

The initial values of IDR1 after a reset and in standby mode are undetermined.

18.2.5 Output Data Register 1 (ODR)

Bit	7	6	5	4	3	2	1
	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1
Initial value	—	—	—	—	—	—	—
Slave Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Host Read/Write	R	R	R	R	R	R	R

ODR1 is an 8-bit readable/writable register to the slave processor, and an 8-bit read-on
to the host processor. The ODRn contents are output on the host data bus when HA0 is
(n = 1 to 4) is low, and $\overline{IOR}$ is low.

The initial values of ODR1 after a reset and in standby mode are undetermined.

Note: * Only 0 can be written, to clear the flag.

STRn (n = 1 to 4) is an 8-bit register that indicates status information during host inter processing. Bits 3, 1, and 0 are read-only bits to both the host and slave processors.

STR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 4 and Bit 2—Defined by User (DBU): The user can use these bits as necess

Bit 3—Command/Data (C/$\overline{D}$): Receives the HA0 input when the host processor writ and indicates whether IDR1 contains data or a command.

Bit 3

C/$\overline{D}$	Description	
0	Contents of input data register (IDR1) are data	(
1	Contents of input data register (IDR1) are a command	

Bit 1—Input Buffer Full (IBF): Set to 1 when the host processor writes to IDR1. Th internal interrupt source to the slave processor. IBF is cleared to 0 when the slave proc IDR1.

The IBF flag setting and clearing conditions are different when the fast A20 gate is us details see table 18.7.

Bit 1

IBF	Description	
0	[Clearing condition]	
	When the slave processor reads IDR	(
1	[Setting condition]	
	When the host processor writes to IDR	

Table 18.3 shows the conditions for setting and clearing the STR flags.

Table 18.3 Set/Clear Timing for STR Flags

Flag	Setting Condition	Clearing Condition
C/$\overline{D}$	Rising edge of host's write signal ($\overline{IOW}$) when HA0 is high	Rising edge of host's write signal ($\overline{IO}$ HA0 is low
IBF*	Rising edge of host's write signal ($\overline{IOW}$) when writing to IDR1	Falling edge of slave's internal read when reading IDR1
OBF	Falling edge of slave's internal write signal ($\overline{WR}$) when writing to ODR1	Rising edge of host's read signal ($\overline{IO}$ reading ODR1

Note: * The IBF flag setting and clearing conditions are different when the fast A20 g used. For details see table 18.7.

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mo
When the MSTP2 bit is set to 1, the host interface halts and enters module stop mode.
25.5, Module Stop Mode, for details.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not in
software standby mode.

MSTPCRL Bit 2—Module Stop (MSTP2): Specifies host interface module stop mo

MSTPCRL Bit 2	
MSTP2	**Description**
0	Host interface module stop mode is cleared
1	Host interface module stop mode is set (I

18.3 Operation

18.3.1 Host Interface Activation

The HIF (slave mode) is activated by setting the HI12E bit (bit 0) in SYSCR2 to 1 in s
mode. When the HIF (slave mode) is activated, all related I/O ports (data port 3, contr
and 9, and host interrupt request port 4) become dedicated host interface ports. Setting
bit and CS4E bit to 1 enables the number of HIF channels to be extended to a four, an
channel 3 and 4 related I/O port (part of port B for control and host interrupt requests)
host interface port.

Table 18.4 shows HIF host interface channel selection and pin operation.

		1	Host interface channel 1 and 4 functions operating Operation of channels 2 and 3 halted (No operation as $\overline{CS2}$ or $\overline{ECS2}$ and $\overline{CS3}$ inputs. Pins P90, PB0, and PB2 operate as I/O ports.)
	1	0	Host interface channel 1 and 3 functions operating Operation of channels 2 and 4 halted (No operation as $\overline{CS2}$ or $\overline{ECS2}$ and $\overline{CS4}$ inputs. Pins P90, PB1, and PB3 operate as I/O ports.)
		1	Host interface channel 1, 3, and 4 functions operatin Operation of channel 2 halted (No operation as $\overline{CS2}$ or $\overline{ECS2}$ input. Pins P43, P81 operate as I/O ports.)
1	0	0	Host interface channel 1 and 2 functions operating Operation of channels 3 and 4 halted (No operation as $\overline{CS3}$ and $\overline{CS4}$ inputs. Pins PB0 to P operate as I/O ports.)
		1	Host interface channel 1, 2, and 4 functions operatin Operation of channel 3 halted (No operation as $\overline{CS3}$ input. Pins PB0 and PB2 oper ports.)
	1	0	Host interface channel 1 to 3 functions operating Operation of channel 4 halted (No operation as $\overline{CS4}$ input. Pins PB1 and PB3 oper ports.)
		1	Host interface channel 1 to 4 functions operating

For host read/write timing, see section 26.7.5, Timing of On-Chip Supporting Modules

		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

18.3.3 A20 Gate

The A20 gate signal can mask address A20 to emulate an addressing mode used by pe computers with an 8086*-family CPU. In slave mode, a regular-speed A20 gate signa output under firmware control. Fast A20 gate output is enabled by setting the FGA20E to 1 in HICR (H'FFF0).

Note: * Intel microprocessor.

Regular A20 Gate Operation

Output of the A20 gate signal can be controlled by an H'D1 command followed by dat slave processor receives data, it normally uses an interrupt routine activated by the IBF to read IDR1. If the data follows an H'D1 command, software copies bit 1 of the data 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 a fast A20 gate sign P81DDR must be set to 1 to assign this pin for output. The initial output from this pin logic 1, which is the initial value. Afterward, the host processor can manipulate the ou

Table 18.6 GA20 (P81) Set/Clear Timing

Pin Name	Setting Condition	Clearing Condition
GA20 (P81)	Rising edge of the host's write signal ($\overline{IOW}$) 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 ($\overline{IOW}$) when bit 1 of the written and the data follows an H'D1 h command Also, when bit FGA20E in HICF to 0

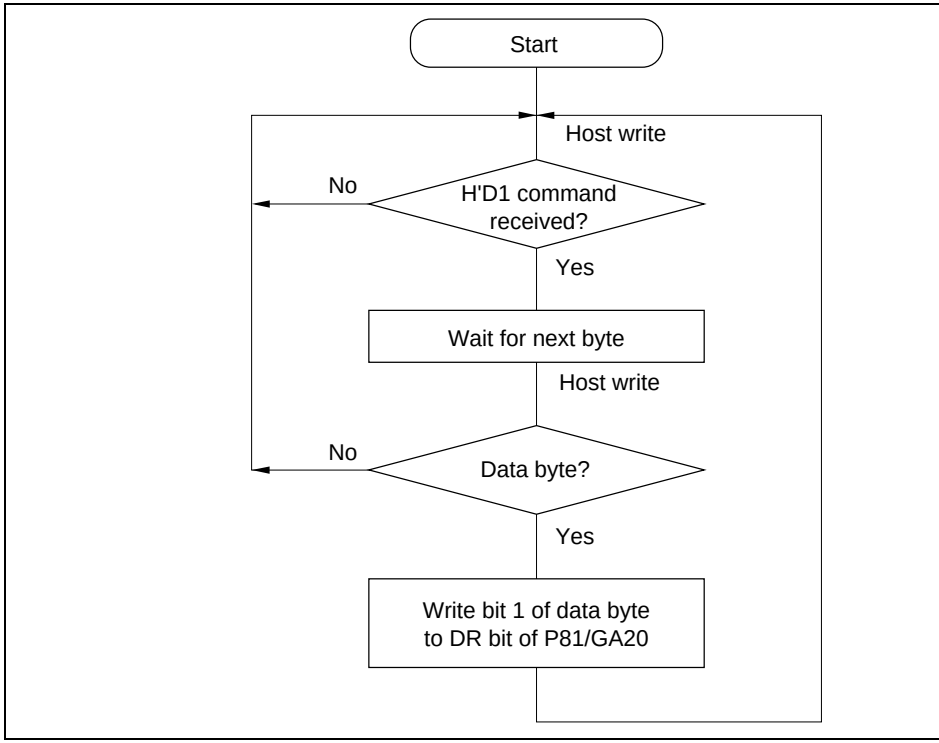


Figure 18.2 GA20 Output

1	H'FF command	0	Q (0)	
1	H'D1 command	0	Q	Turn-on seque
0	1 data*1	0	1	(abbreviated fo
1/0	Command other than H'FF and H'D1	1	Q (1)	
1	H'D1 command	0	Q	Turn-off seque
0	0 data*2	0	0	(abbreviated fo
1/0	Command other than H'FF and H'D1	1	Q (0)	
1	H'D1 command	0	Q	Cancelled sequ
1	Command other than H'D1	1	Q	
1	H'D1 command	0	Q	Retriggered se
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.

18.3.4 Host Interface Pin Shutdown Function

Host interface output can be placed in the high-impedance state according to the state HIFSD pin. Setting the SDE bit to 1 in the SYSCR2 register enables the HIFSD pin is The HIF constantly monitors the HIFSD pin, and when this pin goes low, places the h output pins (HIRQ1, HIRQ11, HIRQ12, HIRQ3, HIRQ4, and GA20) in the high-impe At the same time, the host interface input pins ($\overline{CS1}$, $\overline{CS2}$ or $\overline{ECS2}$, $\overline{CS3}$, $\overline{CS4}$, $\overline{IOW}$, $\overline{I}$ HA0) are disabled (fixed at the high input state internally) regardless of the pin states, signals of the multiplexed functions of these pins (input block) are similarly fixed inte result, the host interface I/O pins (HDB7 to HDB0) also go to the high-impedance stat

This state is maintained while the HIFSD pin is low, and when the HIFSD pin returns level state, the pins are restored to their normal operation as host interface pins.

$\overline{CS1}$	P95	O	Input	Slave mode
$\overline{CS2}$	P81	Δ	Input	Slave mode and CS2E = 1 and FG
$\overline{ECS2}$	P90	Δ	Input	Slave mode and CS2E = 1 and FG
$\overline{CS3}$	PB2	Δ	Input	Slave mode and CS3E = 1
$\overline{CS4}$	PB3	Δ	Input	Slave mode and CS4E = 1
HA0	P80	O	Input	Slave mode
HDB7 to HDB0	P37 to P30	O	I/O	Slave mode
HIRQ11	P43	Δ	Output	Slave mode and CS2E = 1 and P43
HIRQ1	P44	Δ	Output	Slave mode and P44DDR = 1
HIRQ12	P45	Δ	Output	Slave mode and P45DDR = 1
HIRQ3	PB0	Δ	Output	Slave mode and CS3E = 1 and PB
HIRQ4	PB1	Δ	Output	Slave mode and CS4E = 1 and PB
GA20	P81	Δ	Output	Slave mode and FGA20E = 1
HIFSD	P82	—	Input	Slave mode and SDE = 1

Legend:

O: Pins shut down by shutdown function

The $\overline{IRQ2}/\overline{ADTRG}$ input signal is also fixed in the case of P90 shutdown, the TMCI1 signal in the case of P43 shutdown, and the TMRI/CSYNCI in the case of P45 shut

Δ: Pins shut down only when the HIF function is selected by means of a register settin

—: Pin not shut down

Note: Slave mode: Single-chip mode and HI12E = 1

Table 18.9 Input Buffer Full Interrupts

Interrupt	Description
IBF1	Requested when IBFIE1 is set to 1 and IDR1 is full
IBF2	Requested when IBFIE2 is set to 1 and IDR2 is full
IBF3	Requested when IBFIE3 is set to 1 and IDR3 is full
IBF4	Requested when IBFIE4 is set to 1 and IDR4 is full

18.4.2 HIRQ11, HIRQ1, HIRQ12, HIRQ3, and HIRQ4

In slave mode (single-chip mode, with HI12E = 1 in SYSCR2), bits P45DR to P43DR
4 data register (P4DR) and bits PB1ODR and PB0ODR in the port B data register (PB
be used as host interrupt request latches

The corresponding bits in P4DR are cleared to 0 by the host processor's read signal ($\overline{I}$
and HA0 are low, when $\overline{IOR}$ goes low and the host reads ODR1, HIRQ1 and HIRQ12
to 0. If $\overline{CS2}$ and HA0 are low, when $\overline{IOR}$ goes low and the host reads ODR2, HIRQ11
0. The corresponding bit in PBODR is cleared to 0 by the host's read signal ($\overline{IOR}$). If
HA0 are low, when $\overline{IOR}$ goes low and the host reads ODR3, HIRQ3 is cleared to 0. If
HA0 are low, when $\overline{IOR}$ goes low and the host reads ODR4, HIRQ4 is cleared to 0. T
host interrupt request, normally on-chip firmware writes 1 in the corresponding bit. In
the interrupt, the host's interrupt handling routine reads the output data register (ODR
ODR3, or ODR4) and this clears the host interrupt latch to 0.

Table 18.10 indicates how these bits are set and cleared. Figure 18.3 shows the proces
flowchart form.

(P45)	then writes 1	host reads output data regi
HIRQ3 (PB0)	Internal CPU reads 0 from bit PB0ODR, then writes 1	Internal CPU writes 0 in bit or host reads output data r
HIRQ4 (PB1)	Internal CPU reads 0 from bit PB1ODR, then writes 1	Internal CPU writes 0 in bit or host reads output data r

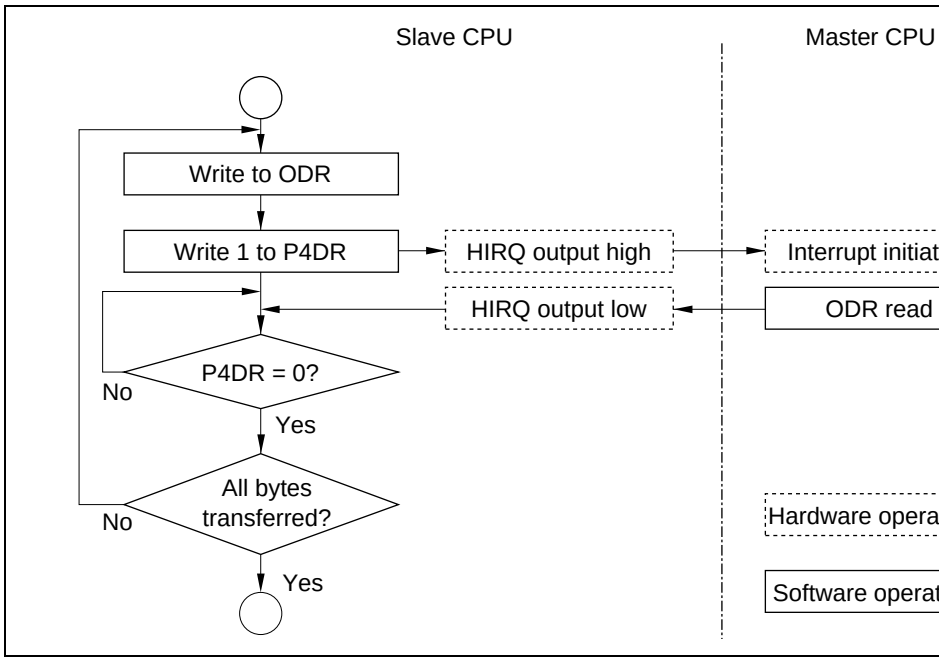


Figure 18.3 HIRQ Output Flowchart (Example of Channels 1 and 2)

The following points require attention when using the host interface.

(1) Host and slave transmission/reception procedures

The host interface provides buffering of asynchronous data from the host and slav but an interface protocol must be followed to implement necessary functions and a contention. For example, if the host and slave processors try to access the same inp data register simultaneously, the data will be corrupted. Interrupts can be used to d simple and effective protocol.

(2) Preventing data contention on the HDB

When the HIF function is used (HI12E = 1 in SYSCR2) and channel 3 or channel set as deselected (CS3E = 0 or CS4E = 0 in SYSCR2), apply either of the followin conditions.

1. Ensure that the $\overline{CS}$ pin for the deselected channel is fixed high.
2. Do not perform port B reads.

(3) Preventing through-current in pins $\overline{CS1}$ to $\overline{CS4}$

Also, if two or more of pins $\overline{CS1}$ to $\overline{CS4}$ are driven low simultaneously in attempti ODR access, signal contention will occur within the chip, and a through-current m This usage must therefore be avoided.

Features of the D/A converter module are listed below.

- Eight-bit resolution
- Two-channel output
- Maximum conversion time: 10 µs (with 20-pF load capacitance)
- Output voltage: 0 V to AV_{ref}
- D/A output retention in software standby mode

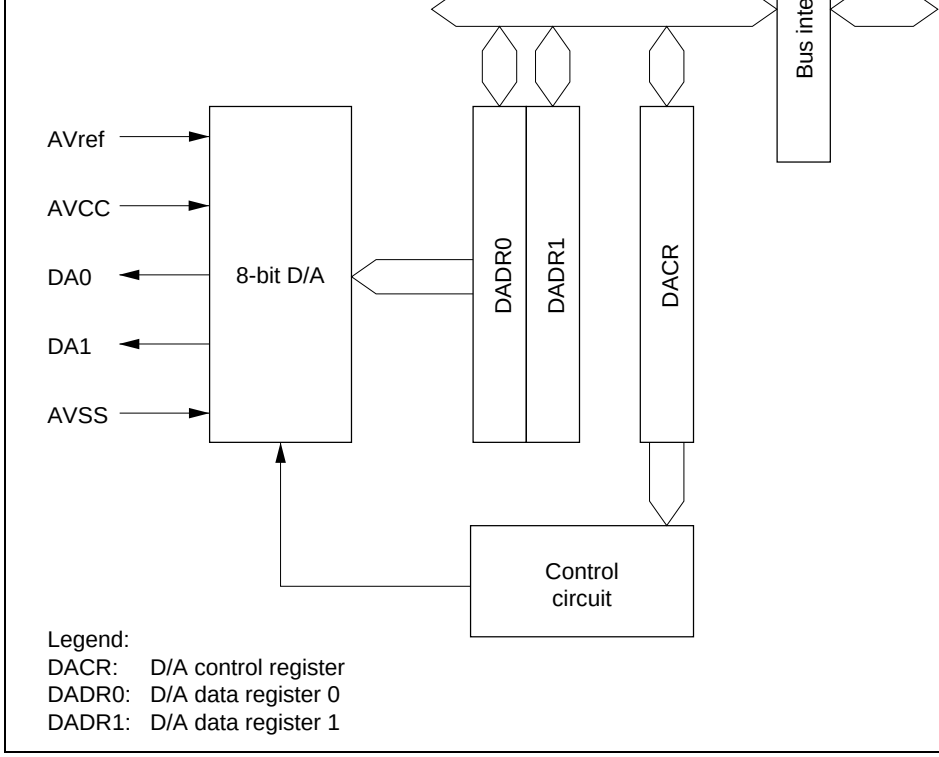


Figure 19.1 Block Diagram of D/A Converter

			circuits
Analog output 0	DA0	Output	Analog output channel 0
Analog output 1	DA1	Output	Analog output channel 1
Reference voltage pin	AVref	Input	Reference voltage for analog cir

19.1.4 Register Configuration

Table 19.2 lists the registers of the D/A converter module.

Table 19.2 D/A Converter Registers

Name	Abbreviation	R/W	Initial Value	Add
D/A data register 0	DADR0	R/W	H'00	H'FF
D/A data register 1	DADR1	R/W	H'00	H'FF
D/A control register	DACR	R/W	H'1F	H'FF
Module stop control register	MSTPCRH	R/W	H'3F	H'FF
	MSTPCRL	R/W	H'FF	H'FF

Note: * Lower 16 bits of the address.

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

D/A data registers 0 and 1 (DADR0 and DADR1) are 8-bit readable/writable registers t
data to be converted. When analog output is enabled, the value in the D/A data register
converted and output continuously at the analog output pin.

The D/A data registers are initialized to H'00 by a reset and in hardware standby mode.

19.2.2 D/A Control Register (DACR)

Bit	7	6	5	4	3	2	1
	DAOE1	DAOE0	DAE	—	—	—	—
Initial value	0	0	0	1	1	1	1
Read/Write	R/W	R/W	R/W	—	—	—	—

DACR is an 8-bit readable/writable register that controls the operation of the D/A conv
module.

DACR is initialized to H'1F by a reset and in hardware standby mode.

Bit 7—D/A Output Enable 1 (DAOE1): Controls D/A conversion and analog output.

Bit 7

DAOE1	Description
0	Analog output DA1 is disabled (In
1	D/A conversion is enabled on channel 1. Analog output DA1 is enabled

Bit 5—D/A Enable (DAE): Controls D/A conversion, in combination with bit DAOE1. D/A conversion is controlled independently on channels 0 and 1 when DAE = 0. Channels 0 and 1 are controlled together when DAE = 1.

Output of the converted results is always controlled independently by DAOE0 and DA

Bit 7	Bit 6	Bit 5	
DAOE1	**DAOE0**	**DAE**	**D/A conversion**
0	0	*	Disabled on channels 0 and 1
	1	0	Enabled on channel 0 Disabled on channel 1
		1	Enabled on channels 0 and 1
1	0	0	Disabled on channel 0 Enabled on channel 1
		1	Enabled on channels 0 and 1
	1	*	Enabled on channels 0 and 1

Legend:

*: Don't care

If the chip enters software standby mode while D/A conversion is enabled, the D/A ou retained and the analog power supply current is the same as during D/A conversion. If necessary to reduce the analog power supply current in software standby mode, disabl output by clearing the DAOE0, DAOE1, and DAE bits to 0.

Bits 4 to 0—Reserved: These bits cannot be modified and are always read as 1.

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mod
When the MSTP10 bit is set to 1, the D/A converter halts and enters module stop mode
of the bus cycle. See section 25.5, Module Stop Mode, for details.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini
software standby mode.

MSTPCRH Bit 2—Module Stop (MSTP10): Specifies D/A converter module stop m

MSTPCRH Bit 2	
MSTP10	**Description**
0	D/A converter module stop mode is cleared
1	D/A converter module stop mode is set (In

- Software writes the data to be converted in DADR0.
- D/A conversion begins when the DAOE0 bit in DACR is set to 1. After the elapse conversion time, analog output appears at the DA0 pin. The output value is AVref value)/256.

  This output continues until a new value is written in DADR0 or the DAOE0 bit is
- If a new value is written in DADR0, conversion begins immediately. Output of the result begins after the conversion time.
- When the DAOE0 bit is cleared to 0, DA0 becomes an input pin.

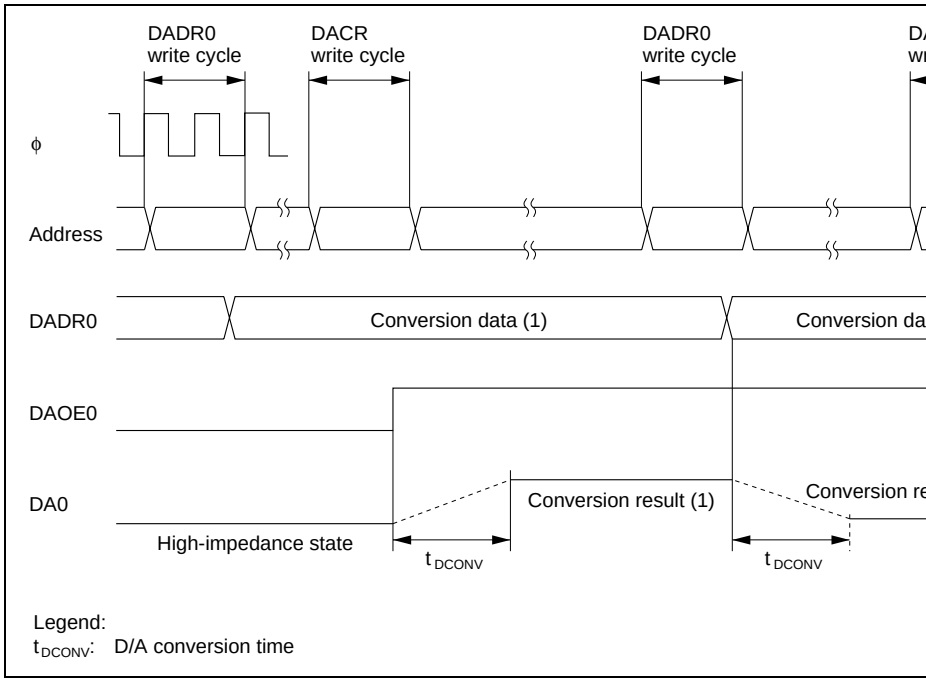


Figure 19.2 D/A Conversion (Example)

In addition to the eight analog input channels, up to 16 channels of digital input can be
A/D conversion. Since the conversion precision falls when digital input is selected, dig
ideal for use by a comparator identifying multi-valued inputs, for example.

20.1.1 Features

A/D converter features are listed below.

- 10-bit resolution
- Eight (analog) or 16 (digital) input channels
- Settable analog conversion voltage range
  - The analog conversion voltage range is set using the reference power supply vo
    (AVref) as the analog reference voltage
- High-speed conversion
  - Minimum conversion time: 6.7 µs per channel (at 20-MHz operation)
- Choice of single mode or scan mode
  - 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
  - Choice of software or timer conversion start trigger (8-bit timer), or $\overline{\text{ADTRG}}$ p
- A/D conversion end interrupt generation
  - An A/D conversion end interrupt (ADI) request can be generated at the end of
    conversion

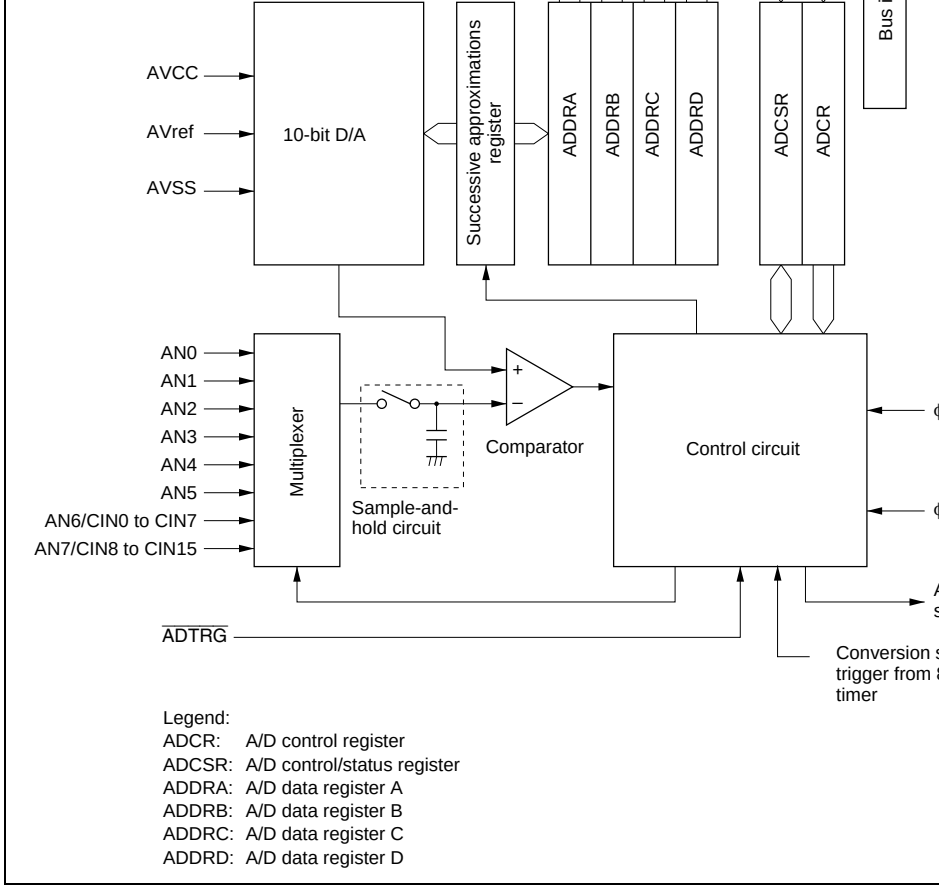


Figure 20.1 Block Diagram of A/D Converter

Analog power supply pin	AVCC	Input	Analog block power supply
Analog ground pin	AVSS	Input	Analog block ground and A/D reference voltage
Reference power supply pin	AVref	Input	A/D conversion reference volta
Analog input pin 0	AN0	Input	Analog input channel 0
Analog input pin 1	AN1	Input	Analog input channel 1
Analog input pin 2	AN2	Input	Analog input channel 2
Analog input pin 3	AN3	Input	Analog input channel 3
Analog input pin 4	AN4	Input	Analog input channel 4
Analog input pin 5	AN5	Input	Analog input channel 5
Analog input pin 6	AN6	Input	Analog input channel 6
Analog input pin 7	AN7	Input	Analog input channel 7
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input for startin conversion
Expansion A/D input pins 0 to 15	CIN0 to CIN15	Input	Expansion A/D conversion inp input pin) channels 0 to 15

A/D data register BH	ADDRBH	R	H'00	H'
A/D data register BL	ADDRBL	R	H'00	H'
A/D data register CH	ADDRCH	R	H'00	H'
A/D data register CL	ADDRCL	R	H'00	H'
A/D data register DH	ADDRDH	R	H'00	H'
A/D data register DL	ADDRDL	R	H'00	H'
A/D control/status register	ADCSR	R/(W)*2	H'00	H'
A/D control register	ADCR	R/W	H'3F	H'
Module stop control register	MSTPCRH	R/W	H'3F	H'
	MSTPCRL	R/W	H'FF	H'
Keyboard comparator control register	KBCOMP	R/W	H'00	H'

Notes: 1. Lower 16 bits of the address.
2. Only 0 can be written in bit 7, to clear the flag.

20.2 Register Descriptions

20.2.1 A/D Data Registers A to D (ADDRA to ADDRD)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R	R	R	R	R	R	R

There are four 16-bit read-only ADDR registers, ADDRA to ADDRD, used to store the A/D conversion.

the lower byte, data transfer is performed via a temporary register (TEMP). For details 20.3, Interface to Bus Master.

The ADDR registers are initialized to H'0000 by a reset, and in standby mode, watch r subactive mode, subsleep mode, and module stop mode.

Table 20.3 Analog Input Channels and Corresponding ADDR Registers

Analog Input Channel		
Group 0	**Group 1**	**A/D Data Register**
AN0	AN4	ADDRA
AN1	AN5	ADDRB
AN2	AN6 or CIN0 to CIN7	ADDRC
AN3	AN7 or CIN8 to CIN15	ADDRD

20.2.2 A/D Control/Status Register (ADCSR)

Bit	7	6	5	4	3	2	1
	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Only 0 can be written in bit 7, to clear the flag.

ADCSR is an 8-bit readable/writable register that controls A/D conversion operations.

ADCSR is initialized to H'00 by a reset, and in standby mode, watch mode, subactive subsleep mode, and module stop mode.

- Single mode: When A/D conversion ends
- Scan mode: When A/D conversion ends on all specified channels

Bit 6—A/D Interrupt Enable (ADIE): Selects enabling or disabling of interrupt (ADI at the end of A/D conversion.

Bit 6

ADIE	Description
0	A/D conversion end interrupt (ADI) request is disabled
1	A/D conversion end interrupt (ADI) request is enabled

Bit 5—A/D Start (ADST): Selects starting or stopping of A/D conversion. Holds a val during A/D conversion.

The ADST bit can be set to 1 by software, a timer conversion start trigger, or the A/D e trigger input pin ($\overline{\text{ADTRG}}$).

Bit 5

ADST	Description
0	A/D conversion stopped
1	Single mode: A/D conversion is started. Cleared to 0 automatically when c on the specified channel ends Scan mode: A/D conversion is started. Conversion continues sequentially selected channels until ADST is cleared to 0 by software, a r transition to standby mode or module stop mode

Bit 4—Scan Mode (SCAN): Selects single mode or scan mode as the A/D conversion mode. See section 20.4, Operation, for single mode and scan mode operation. Only set bit while conversion is stopped.

Bit 3	
CKS	**Description**
0	Conversion time = 266 states (max.)
1	Conversion time = 134 states (max.)

Bits 2 to 0—Channel Select 2 to 0 (CH2 to CH0): Together with the SCAN bit, thes the analog input channel(s).

One analog input channel can be switched to digital input.

Only set the input channel while conversion is stopped.

Group Selection	Channel Selection		Description	
CH2	**CH1**	**CH0**	**Single Mode**	**Scan Mode**
0	0	0	AN0 (Initial value)	AN0
		1	AN1	AN0, AN1
	1	0	AN2	AN0 to AN2
		1	AN3	AN0 to AN3
1	0	0	AN4	AN4
		1	AN5	AN4, AN5
	1	0	AN6 or CIN0 to CIN7	AN4, AN5, AN6 or CIN
		1	AN7 or CIN8 to CIN15	AN4, AN5, AN6 or CIN AN7 or CIN

conversion operations.

ADCR is initialized to H'3F by a reset, and in standby mode, watch mode, subactive m
subsleep mode, and module stop mode.

Bits 7 and 6—Timer Trigger Select 1 and 0 (TRGS1, TRGS0): These bits select ena
disabling of the start of A/D conversion by a trigger signal. Only set bits TRGS1 and T
while conversion is stopped.

Bit 7	Bit 6	
TRGS1	**TRGS0**	**Description**
0	0	Start of A/D conversion by external trigger is disabled (I
	1	Start of A/D conversion by external trigger is disabled
1	0	Start of A/D conversion by external trigger (8-bit timer) is enabl
	1	Start of A/D conversion by external trigger pin is enabled

Bits 5 to 0—Reserved: Should always be written with 1.

Note: Some of these bits are readable/writable in products other than the HD64F2148
HD64F2147N, HD64F2144, HD64F2142R and HD6432142, however, when w
sure to write 1 here for software compatibility.

KBCOMP is an 8-bit readable/writable register that controls the the CIN input channels for A/D conversion.

KBCOMP is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 4—IrDA Control: See the description in section 15.2.11, Keyboard Compar Register (KBCOMP).

Bit 3—Keyboard A/D Enable (KBADE): Selects either analog input pins (AN6, AN input pins (CIN0 to CIN7, CIN8 to CIN15) for A/D converter channel 6 and channel 7

Bits 2 to 0—Keyboard A/D Channel Select 2 to 0 (KBCH2 to KBCH0): These bits channels for A/D conversion from among the digital input pins. Only set the input cha A/D conversion is stopped.

Bit 3	Bit 2	Bit 1	Bit 0	A/D Converter	A/D Conver
KBADE	KBCH2	KBCH1	KBCH0	Channel 6 Input	Channel 7 I
0	—	—	—	AN6	AN7
1	0	0	0	CIN0	CIN8
			1	CIN1	CIN9
		1	0	CIN2	CIN10
			1	CIN3	CIN11
	1	0	0	CIN4	CIN12
			1	CIN5	CIN13
		1	0	CIN6	CIN14
			1	CIN7	CIN15

MSTPCR, comprising two 8-bit readable/writable registers, performs module stop mod

When the MSTP9 bit in MSTPCR is set to 1, A/D converter operation stops at the end cycle and a transition is made to module stop mode. Registers cannot be read or written module stop mode. For details, see section 25.5, Module Stop Mode.

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

MSTPCRH Bit 1—Module Stop (MSTP9): Specifies the A/D converter module stop

MSTPCRH Bit 1	
MSTP9	**Description**
0	A/D converter module stop mode is cleared
1	A/D converter module stop mode is set (

When reading ADDR, always read the upper byte before the lower byte. It is possible
the upper byte, but if only the lower byte is read, incorrect data may be obtained.

Figure 20.2 shows the data flow for ADDR access.

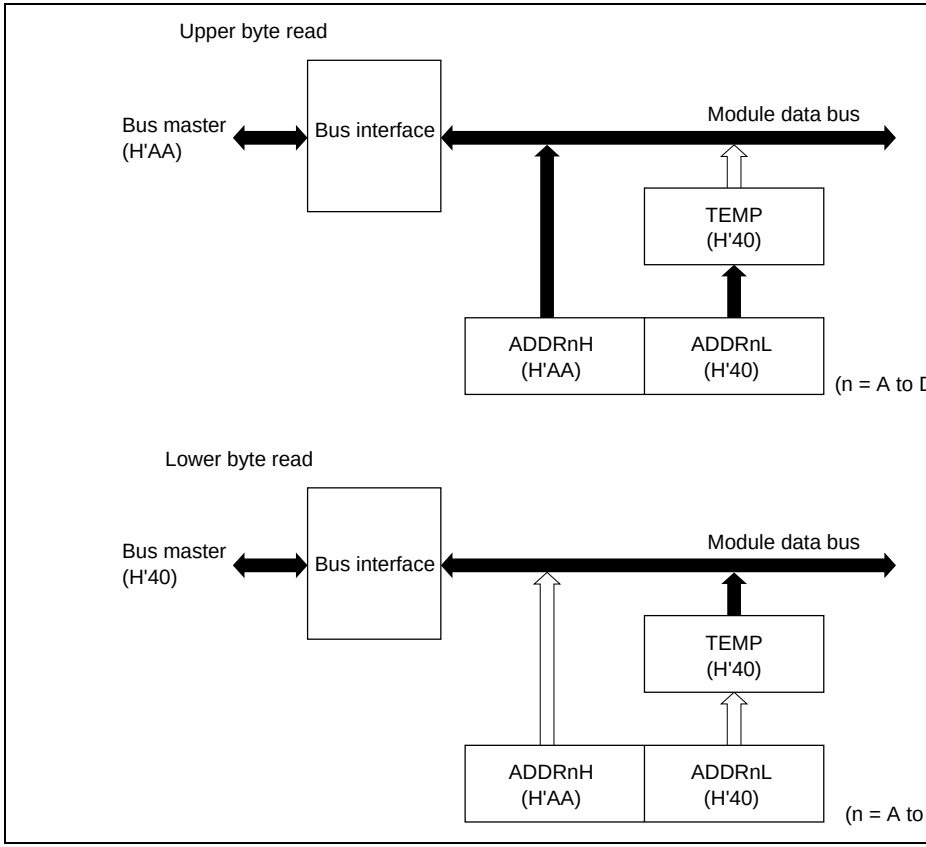


Figure 20.2 ADDR Access Operation (Reading H'AA40)

conversion is started when the ADST bit is set to 1 by software, or by external trigger i
ADST bit remains set to 1 during A/D conversion, and is automatically cleared to 0 wh
conversion ends.

On completion of conversion, the ADF flag is set to 1. If the ADIE bit is set to 1 at this
ADI interrupt request is generated. The ADF flag is cleared by writing 0 after reading A

When the operating mode or analog input channel must be changed during analog conv
prevent incorrect operation, first clear the ADST bit to 0 in ADCSR to halt A/D conver
making the necessary changes, set the ADST bit to 1 to start A/D conversion again. The
can be set at the same time as the operating mode or input channel is changed.

Typical operations when channel 1 (AN1) is selected in single mode are described next
20.3 shows a timing diagram for this example.

1. Single mode is selected (SCAN = 0), input channel AN1 is selected (CH1 = 0, CH0
   A/D interrupt is enabled (ADIE = 1), and A/D conversion is started (ADST = 1).

2. When A/D conversion is completed, the result is transferred to ADDRB. At the sam
   ADF flag is set to 1, the ADST bit is cleared to 0, and the A/D converter becomes id

3. Since ADF = 1 and ADIE = 1, an ADI interrupt is requested.

4. The A/D interrupt handling routine starts.

5. The routine reads ADCSR, then writes 0 to the ADF flag.

6. The routine reads and processes the conversion result (ADDRB).

7. Execution of the A/D interrupt handling routine ends. After that, if the ADST bit is
   A/D conversion starts again and steps 2 to 7 are repeated.

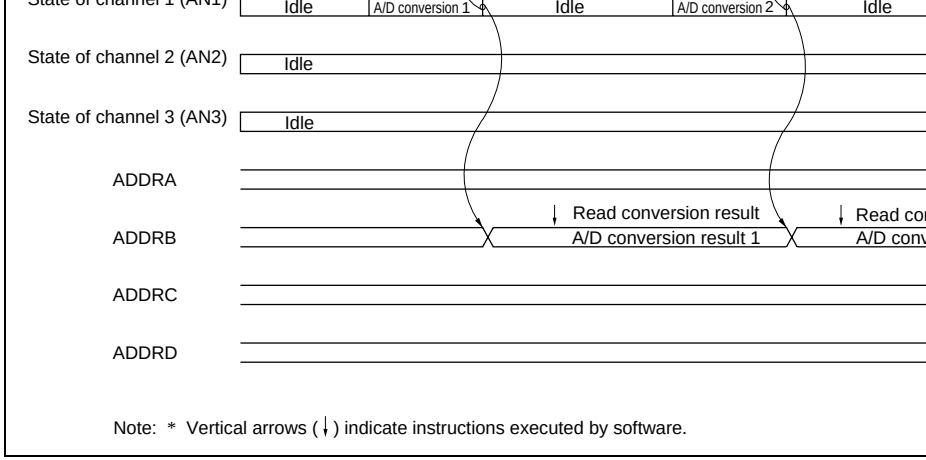


Figure 20.3 Example of A/D Converter Operation (Single Mode, Channel 1 S

registers corresponding to the channels.

When the operating mode or analog input channel must be changed during analog conv
prevent incorrect operation, first clear the ADST bit to 0 in ADCSR to halt A/D conver
making the necessary changes, set the ADST bit to 1 to start A/D conversion again. Th
can be set at the same time as the operating mode or input channel is changed.

Typical operations when three channels (AN0 to AN2) are selected in scan mode are de
next. Figure 20.4 shows a timing diagram for this example.

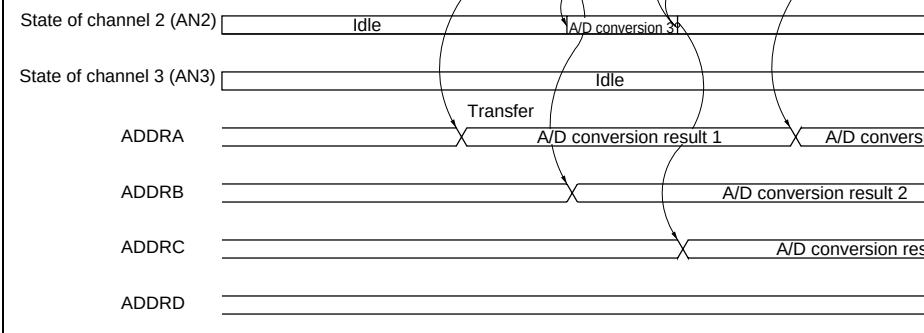
1. Scan mode is selected (SCAN = 1), scan group 0 is selected (CH2 = 0), analog inpu
   AN0 to AN2 are selected (CH1 = 1, CH0 = 0), and A/D conversion is started (ADS

2. When A/D conversion of the first channel (AN0) is completed, the result is transfer
   ADDRA. Next, conversion of the second channel (AN1) starts automatically.

3. Conversion proceeds in the same way through the third channel (AN2).

4. When conversion of all the selected channels (AN0 to AN2) is completed, the ADF
   to 1 and conversion of the first channel (AN0) starts again. If the ADIE bit is set to
   time, an ADI interrupt is requested after A/D conversion ends.

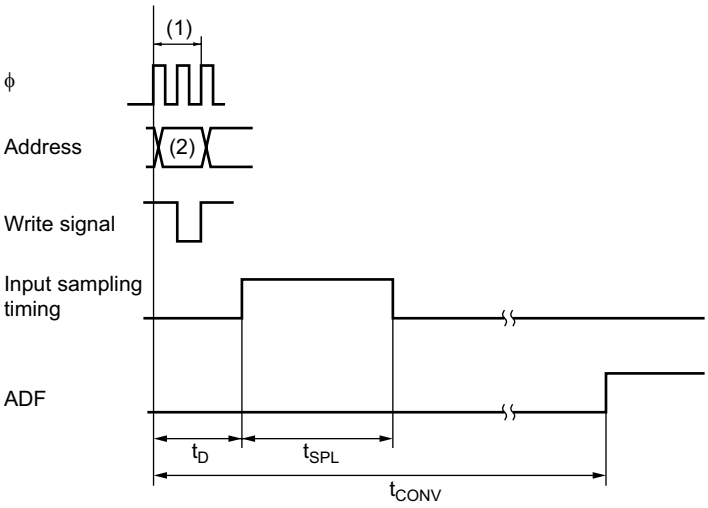
5. Steps 2 to 4 are repeated as long as the ADST bit remains set to 1. When the ADST
   cleared to 0, A/D conversion stops. After that, if the ADST bit is set to 1, A/D conv
   starts again from the first channel (AN0).



Notes: 1. Vertical arrows (↓) indicate instructions executed by software.
2. Data currently being converted is ignored.

Figure 20.4 Example of A/D Converter Operation (Scan Mode, Channels AN0 to AN2 Selected)

In scan mode, the values given in table 20.4 apply to the first conversion time. In the se
subsequent conversions the conversion time is fixed at 256 states when CKS = 0 or 128
when CKS = 1.



Legend:
(1) : ADCSR write cycle
(2) : ADCSR address
t_D : A/D conversion start delay time
t_{SPL} : Input sampling time
t_{CONV} : A/D conversion time

Figure 20.5 A/D Conversion Timing

20.4.4 External Trigger Input Timing

A/D conversion can be externally triggered. When the TRGS1 and TRGS0 bits are set
ADCR, external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge at the $\overline{\text{ADT}}$
the ADST bit to 1 in ADCSR, starting A/D conversion. Other operations, in both sing
modes, are the same as when the ADST bit is set to 1 by software. Figure 20.6 shows

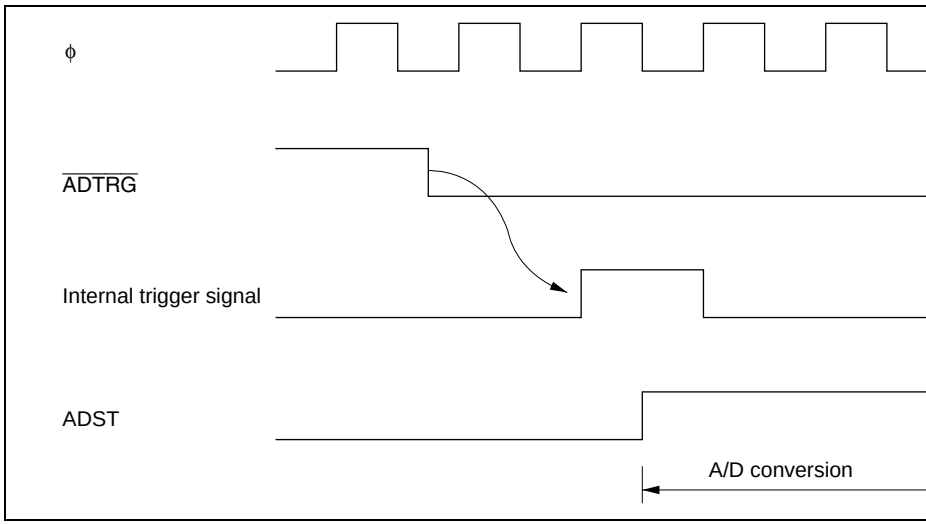


Figure 20.6 External Trigger Input Timing

20.5 Interrupts

The A/D converter generates an interrupt (ADI) at the end of A/D conversion. The AD
request can be enabled or disabled by the ADIE bit in ADCSR.

range $AV_{SS} \leq ANn \leq AV_{ref}$ (n = 0 to 7).

2. Digital input voltage range
   The voltage applied to the CINn digital input pins should be in the range $AV_{SS} \leq CI$ and $V_{SS} \leq CINn \leq V_{CC}$ (n = 0 to 15).

3. Relation between AV_{CC}, AV_{SS} and V_{CC}, V_{SS}
   As the relationship between AV_{CC}, AV_{SS} and V_{CC}, V_{SS}, set $AV_{SS} = V_{SS}$. If the A/D cor not used, the AVCC and AVSS pins must on no account be left open.

4. Setting Range of AVref Pin:
   The reference voltage supplied via the AVref pin should be in the range $AV_{ref} \leq AV$

If conditions 1 to 4 above are not met, the reliability of the device may be adversely aff

Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as and layout in which digital circuit signal lines and analog circuit signal lines cross or ar proximity should be avoided as far as possible. Failure to do so may result in incorrect of the analog circuitry due to inductance, adversely affecting A/D conversion values.

Also, digital circuitry must be isolated from the analog input signals (AN0 to AN7), an reference power supply (AVref), and analog power supply (AVCC) by the analog grou Also, the analog ground (AVSS) should be connected at one point to a stable digital gr on the board.

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) or analog reference power supply pin (AV be connected between AVCC and AVSS as shown in figure 20.7.

when deciding the circuit constants.

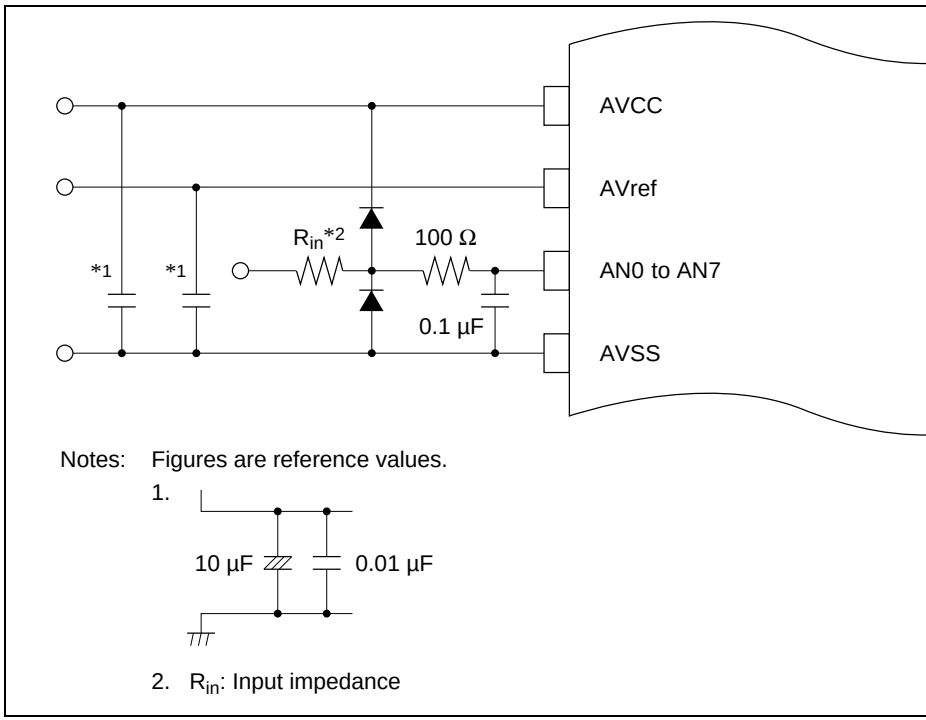


Figure 20.7 Example of Analog Input Protection Circuit

Table 20.5 Analog Pin Ratings

Item	Min	Max	Uni
Analog input capacitance	—	20	pF
Permissible signal source impedance	—	10*	kΩ

Note: * When V_{cc} = 4.0 to 5.5 V and $\phi \leq$ 12 MHz.

A/D Conversion Precision Definitions

The A/D conversion precision in this LSI is defined as follows.

- Resolution
  The number of A/D converter digital output codes

- Offset error
  The deviation of the analog input voltage value from the ideal A/D conversion char
  when the digital output changes from the minimum voltage value B'0000000000 (H
  B'0000000001 (H'001) (see figure 20.10).

- Full-scale error
  The deviation of the analog input voltage value from the ideal A/D conversion char
  when the digital output changes from B'1111111110 (H'3FE) to B'111111111 (H'3F
  figure 20.11).

- Quantization error
  The deviation inherent in the A/D converter, given by 1/2 LSB (see figure 20.9).

- Nonlinearity error
  The error with respect to the ideal A/D conversion characteristic between the zero v
  the full-scale voltage. Does not include the offset error, full-scale error, or quantizat

- Absolute precision
  The deviation between the digital value and the analog input value. Includes the off
  full-scale error, quantization error, and nonlinearity error.

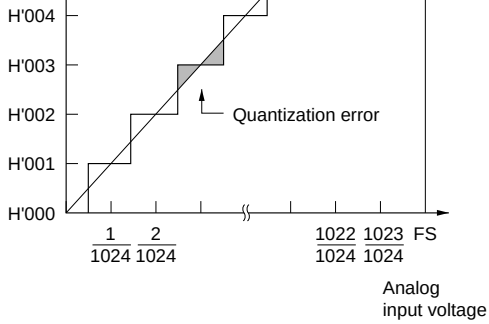


Figure 20.9 A/D Conversion Precision Definitions (1)

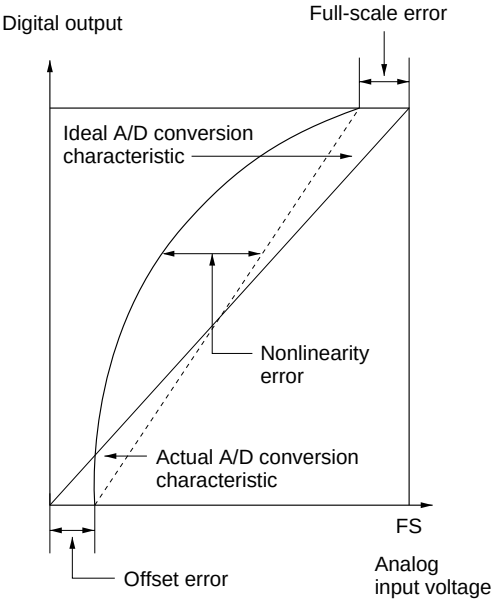


Figure 20.10 A/D Conversion Precision Definitions (2)

However, if a large capacitance is provided externally, the input load will essentially co
only the internal input resistance of 10 kΩ, and the signal source impedance is ignored.

But since a low-pass filter effect is obtained in this case, it may not be possible to follo
signal with a large differential coefficient (e.g., 5 mV/µsec or greater).

When converting a high-speed analog signal, a low-impedance buffer should be inserte

Influences on Absolute Precision

Adding capacitance results in coupling with GND, and therefore noise in GND may ad
affect absolute precision. Be sure to make the connection to an electrically stable GND
AV_{SS}.

Care is also required to insure that filter circuits do not communicate with digital signal
mounting board, so acting as antennas.

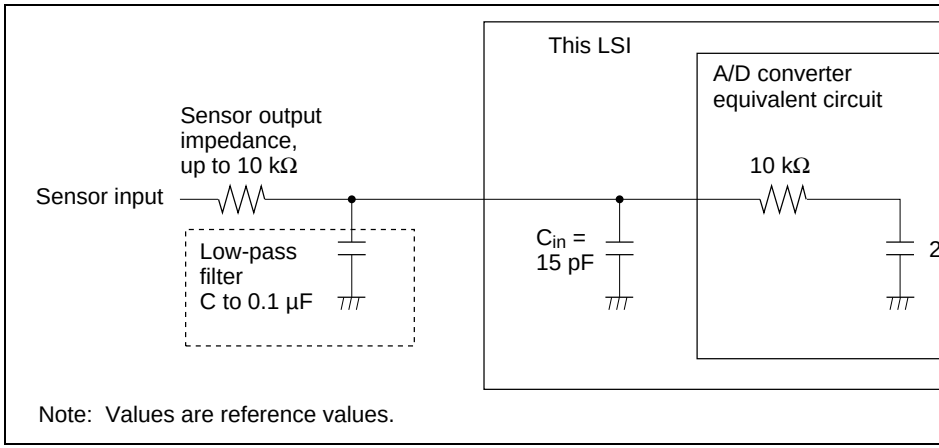


Figure 20.11 Example of Analog Input Circuit

makes it possible to perform fast word data transfer.

The on-chip RAM can be enabled or disabled by means of the RAM enable bit (RAM
system control register (SYSCR).

21.1.1 Block Diagram

Figure 21.1 shows a block diagram of the on-chip RAM.

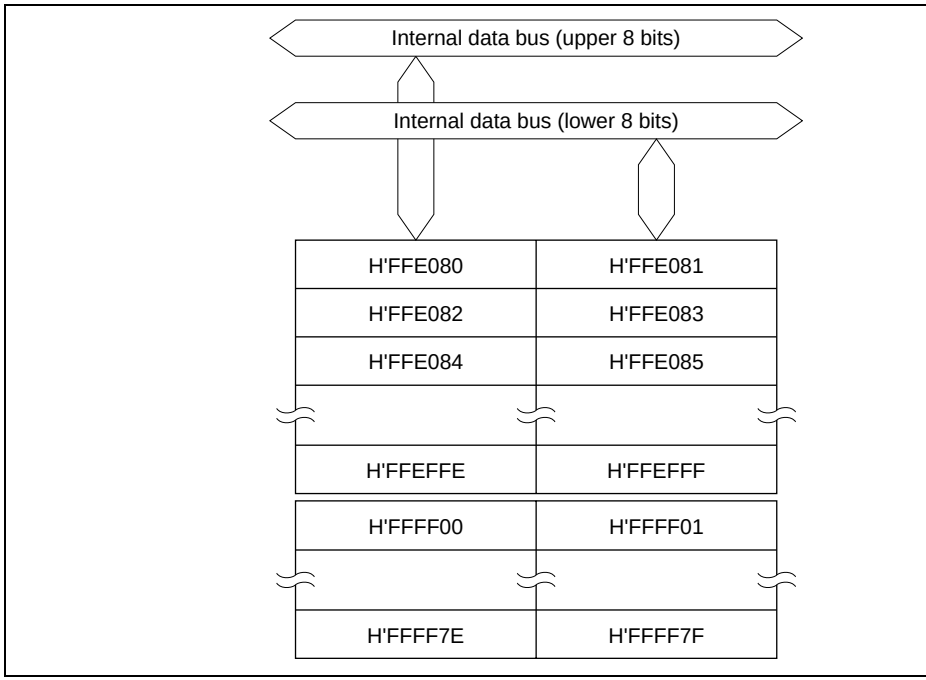


Figure 21.1 Block Diagram of RAM (H8S/2148, H8S/2144, H8S/2143

21.2 System Control Register (SYSCR)

Bit	7	6	5	4	3	2	1
	CS2E	IOSE	INTM1	INTM0	XRST	NMIEG	HIE
Initial value	0	0	0	0	1	0	0
Read/Write	R/W	R/W	R	R/W	R	R/W	R/W

The on-chip RAM is enabled or disabled by the RAME bit in SYSCR. For details of ot
SYSCR, see section 3.2.2, System Control Register (SYSCR).

Bit 0—RAM Enable (RAME): Enables or disables the on-chip RAM. The RAME bit
initialized when the reset state is released. It is not initialized in software standby mode

Bit 0

RAME	Description
0	On-chip RAM is disabled
1	On-chip RAM is enabled (I

H'(FF)EFFF and H'(FF)FF00 to H'(FF)FF7F, are directed to the off-chip address space

Since the on-chip RAM is connected to the bus master by a 16-bit data bus, it can be v
and read in byte or word units. Each type of access is performed in one state.

Even addresses use the upper 8 bits, and odd addresses use the lower 8 bits. Word data
at an even address.

21.3.2 Single-Chip Mode (Modes 2 and 3 (EXPE = 0))

When the RAME bit is set to 1, accesses to H8S/2148, H8S/2144, and H8S/2143 addr
H'(FF)E080 to H'(FF)EFFF and H'(FF)FF00 to H'(FF)FF7F, and H8S/2147, H8S/214
H8S/2142 addresses H'(FF)E880 to H'(FF)EFFF and H'(FF)FF00 to H'(FF)FF7F, are
the on-chip RAM. When the RAME bit is cleared to 0, the on-chip RAM is not access
Undefined values are read from these bits, and writing is invalid.

Since the on-chip RAM is connected to the bus master by a 16-bit data bus, it can be v
and read in byte or word units. Each type of access is performed in one state.

Even addresses use the upper 8 bits, and odd addresses use the lower 8 bits. Word data
at an even address.

The H8S/2148 and H8S/2144 have 128 kbytes of on-chip ROM (flash memory or mas
the H8S/2143 has 96 kbytes, the H8S/2147, H8S/2147N, and H8S/2142 have 64 kbyte
ROM is connected to the CPU by a 16-bit data bus. The CPU accesses both byte and w
one state, enabling faster instruction fetches and higher processing speed.

The mode pins (MD1 and MD0) and the EXPE bit in MDCR can be set to enable or d
on-chip ROM.

The flash memory versions of the H8S/2148, H8S/2147N, H8S/2144, and H8S/2142 c
and programmed on-board as well as with a general-purpose PROM programmer.

22.1.1 Block Diagram

Figure 22.1 shows a block diagram of the ROM.

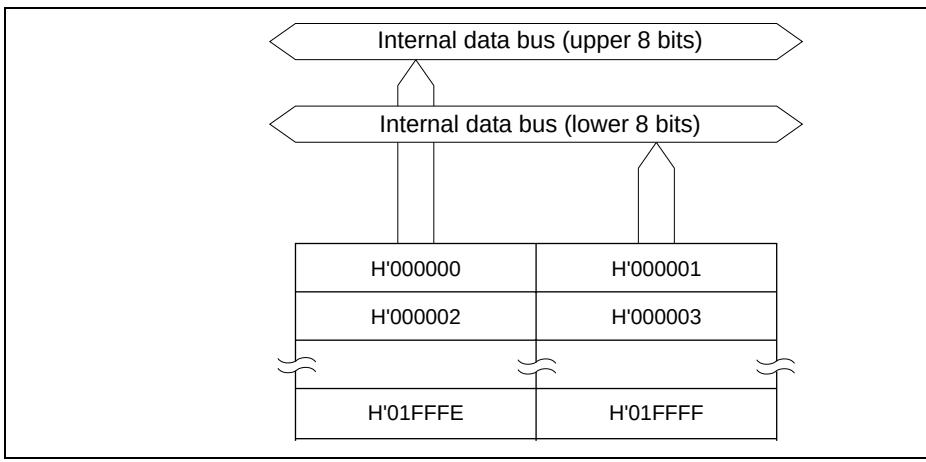


Figure 22.1 ROM Block Diagram (H8S/2148, H8S/2144)

Note: * Lower 16 bits of the address.

22.2 Register Descriptions

22.2.1 Mode Control Register (MDCR)

Bit	7	6	5	4	3	2	1
	EXPE	—	—	—	—	—	MDS1
Initial value	—*	0	0	0	0	0	—*
Read/Write	R/W*	—	—	—	—	—	R

Note: * Determined by the MD1 and MD0 pins.

MDCR is an 8-bit register used to set this group operating mode and monitor the curren mode.

The EXPE bit is initialized in accordance with the mode pin states by a reset and in har standby mode.

Bit 7—Expanded Mode Enable (EXPE): Sets expanded mode. In mode 1, EXPE is f and cannot be modified. In modes 2 and 3, EXPE has an initial value of 0 and can be re written.

Bit 7 EXPE	Description
0	Single-chip mode selected
1	Expanded mode selected

The on-chip ROM is connected to the CPU by a 16-bit data bus, and both byte and wo
accessed in one state. Even addresses are connected to the upper 8 bits, and odd addre
lower 8 bits. Word data must start at an even address.

The mode pins (MD1 and MD0) and the EXPE bit in MDCR can be set to enable or d
on-chip ROM, as shown in table 22.2.

In normal mode, the maximum amount of ROM that can be used is 56 kbytes.

Table 22.2 Operating Modes and ROM

Operating Mode						
MCU Operating Mode	**CPU Operating Mode**	**Description**	**Mode Pins**		**MDCR**	
			MD1	**MD0**	**EXPE**	**On-**
Mode 1	Normal	Expanded mode with on-chip ROM disabled	0	1	1	Disa
Mode 2	Advanced	Single-chip mode	1	0	0	Ena
	Advanced	Expanded mode with on-chip ROM enabled			1	
Mode 3	Normal	Single-chip mode		1	0	Ena
	Normal	Expanded mode with on-chip ROM enabled			1	(ma

Note: * 128 kbytes in the H8S/2148 and H8S/2144, 96 kbytes in the H8S/2143, 64
H8S/2147, H8S/2147N, and H8S/2142.

— Erase mode
— Program-verify mode
— Erase-verify mode

- Programming/erase methods

  The flash memory is programmed 32 bytes at a time. Erasing is performed by block
  single-block units). When erasing multiple blocks, the individual blocks must be era
  sequentially. Block erasing can be performed as required on 1-kbyte, 8-kbyte, 16-kb
  kbyte, and 32-kbyte blocks.
- Programming/erase times

  The flash memory programming time is 10 ms (typ.) for simultaneous 32-byte prog
  equivalent to 300 µs (typ.) per byte, and the erase time is 100 ms (typ.) per block.
- Reprogramming capability

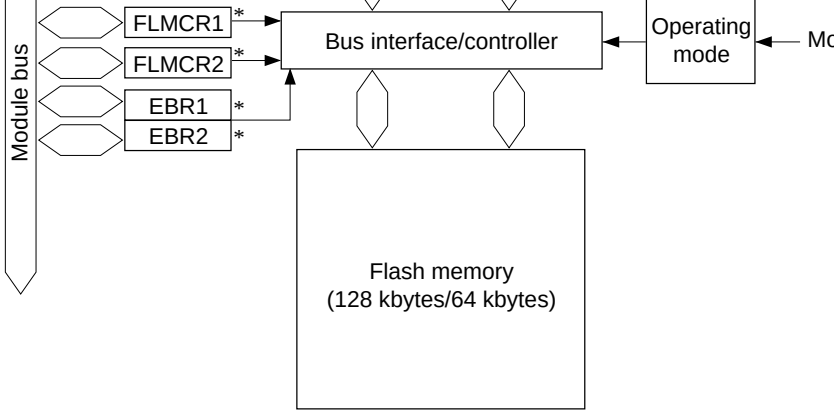
  The flash memory can be reprogrammed up to 100 times.
- On-board programming modes

  There are two modes in which flash memory can be programmed/erased/verified on
  — Boot mode
  — User program mode
- Automatic bit rate adjustment

  With data transfer in boot mode, the bit rate of the chip can be automatically adjuste
  the transfer bit rate of the host.
- Protect modes

  There are three protect modes, hardware, software, and error protect, which allow p
  status to be designated for flash memory program/erase/verify operations.
- Programmer mode

  Flash memory can be programmed/erased in programmer mode, using a PROM pro
  as well as in on-board programming mode.

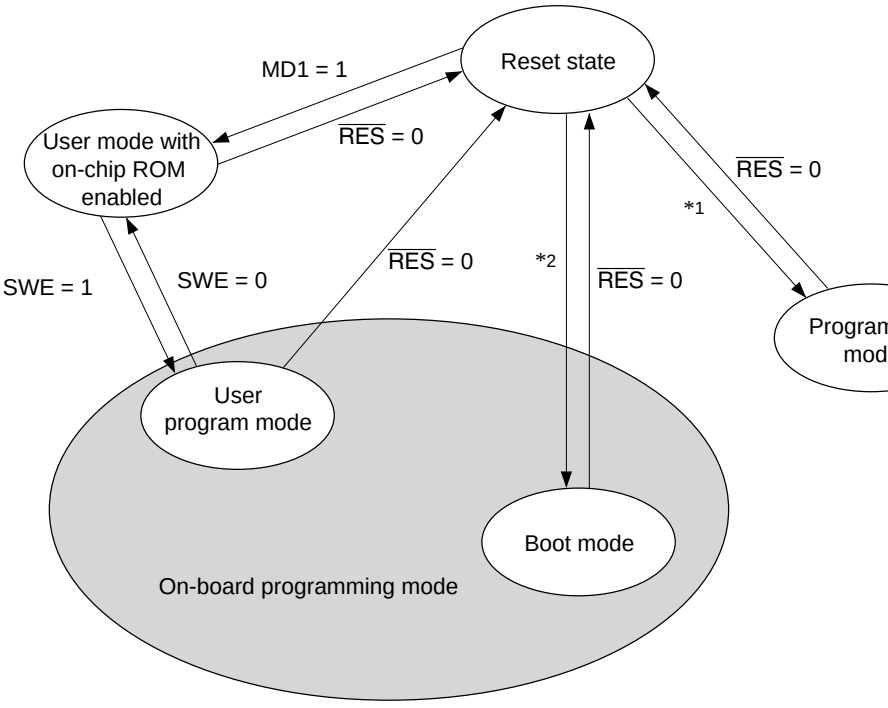


Legend:
FLMCR1: Flash memory control register 1
FLMCR2: Flash memory control register 2
EBR1: Erase block register 1
EBR2: Erase block register 2

Note: * These registers are used only in the flash memory version. In the mask ROM
a read at any of these addresses will return an undefined value, and writes are

Figure 22.2 Block Diagram of Flash Memory

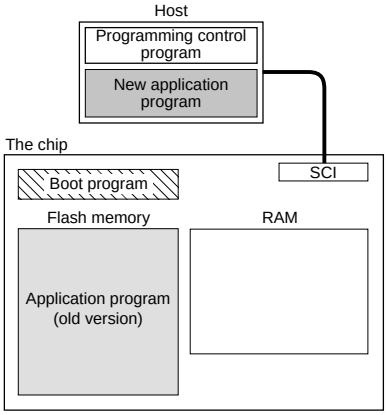
mode.

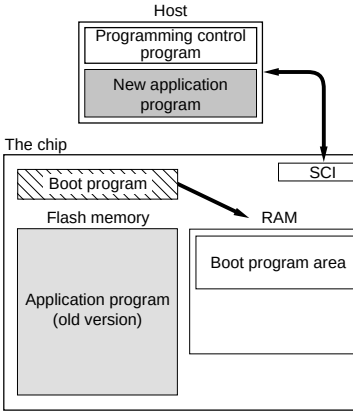


Notes: Only make a transition between user mode and user program mode when the
not accessing the flash memory.

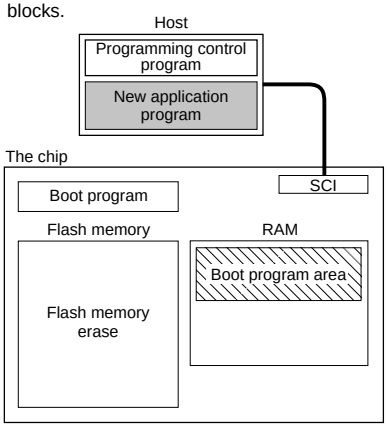
1. MD0 = MD1 = 0, P92 = 0, P91 = P90 = 1
2. MD0 = MD1 = 0, P92 = P91 = P90 = 1

Figure 22.3 Flash Memory Mode Transitions





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, entire flash memory erasure is performed, without regard to blocks.



4. Writing new application program
   The programming control program transfe
   from the host to RAM by SCI communicat
   executed, and the new application progra
   host is written into the flash memory.

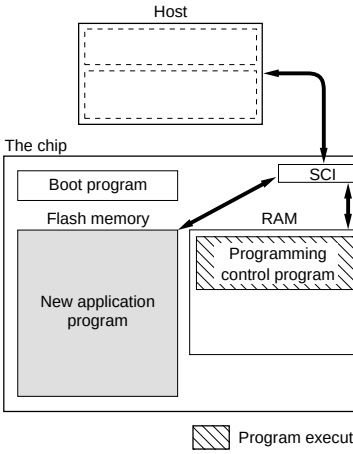
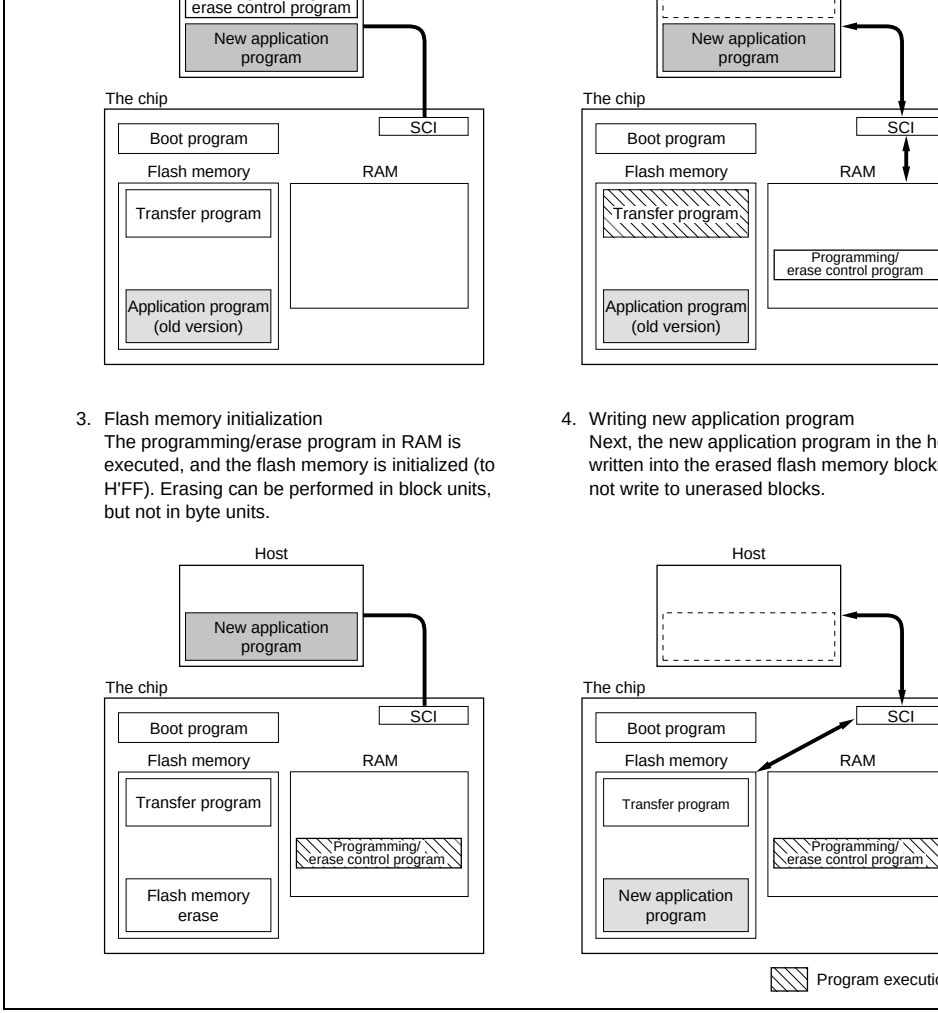


Figure 22.4 Boot Mode



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 h
written into the erased flash memory block
not write to unerased blocks.

Figure 22.5 User Program Mode (Example)

Block Configuration

The flash memory is divided into two 32-kbyte blocks (128-kbyte version only), two 8
blocks, one 16-kbyte block, one 28-kbyte block, and four 1-kbyte blocks.

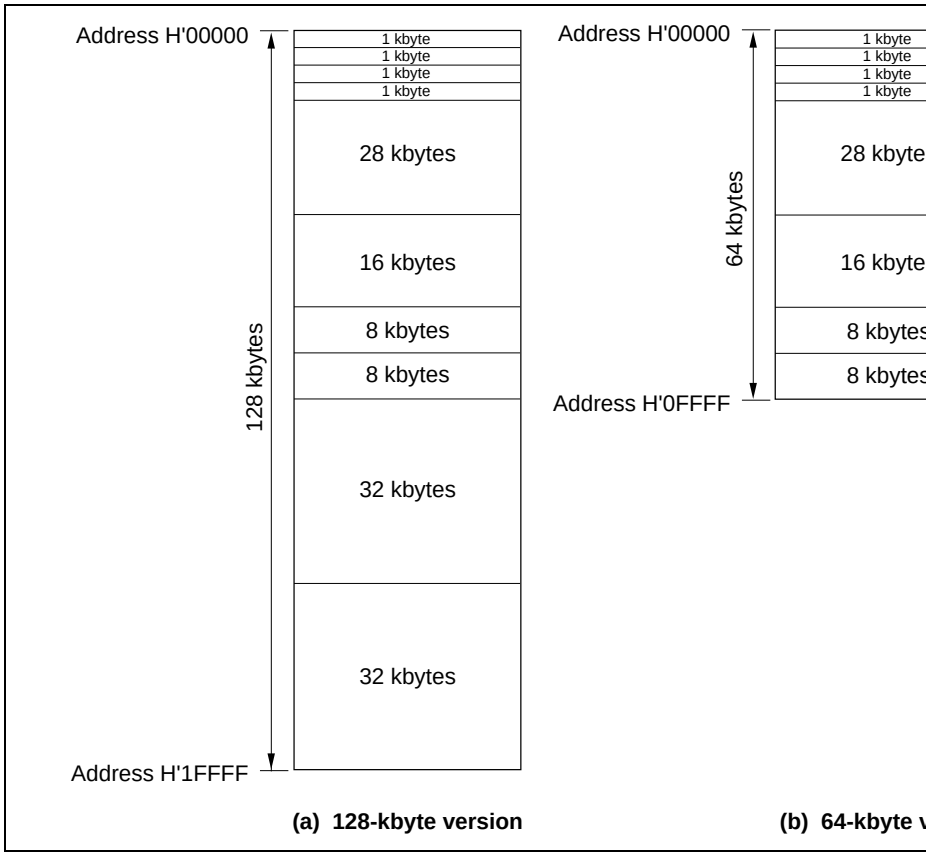


Figure 22.6 Flash Memory Block Configuration

Mode 0	MD0	Input	Sets MCU operating mode
Port 92	P92	Input	Sets MCU operating mode when MD1
Port 91	P91	Input	Sets MCU operating mode when MD1
Port 90	P90	Input	Sets MCU operating mode when MD1
Transmit data	TxD1	Output	Serial transmit data output
Receive data	RxD1	Input	Serial receive data input

22.4.5 Register Configuration

The registers used to control the on-chip flash memory when enabled are shown in tabl
In order for these registers to be accessed, the FLSHE bit must be set to 1 in STCR.

Table 22.4 Flash Memory Registers

Register Name	Abbreviation	R/W	Initial Value	Add
Flash memory control register 1	FLMCR1*5	R/W*3	H'80	H'FI
Flash memory control register 2	FLMCR2*5	R/W*3	H'00*4	H'FI
Erase block register 1	EBR1*5	R/W*3	H'00*4	H'FI
Erase block register 2	EBR2*5	R/W*3	H'00*4	H'FI
Serial/timer control register	STCR	R/W	H'00	H'FI

Notes: 1. Lower 16 bits of the address.
2. Flash memory registers share addresses with other registers. Register selec performed by the FLSHE bit in the serial/timer control register (STCR).
3. In modes in which the on-chip flash memory is disabled, a read will return H' writes are invalid.
4. The SWE bit in FLMCR1 is not set, these registers are initialized to H'00.
5. FLMCR1, FLMCR2, EBR1, and EBR2 are 8-bit registers. Only byte accesse for these registers, the access requiring 2 states. These registers are used o flash memory version. In the mask ROM version, a read at any of these addr return an undefined value, and writes are invalid.

Read/Write	R	R/W	R/W	R/W	R/W	R/W

FLMCR1 is an 8-bit register used for flash memory operating mode control. Program- or erase-verify mode is entered by setting SWE to 1. Program mode is entered by setti 1, then setting the PSU bit in FLMCR2, and finally setting the P bit. Erase mode is en setting SWE to 1, then setting the ESU bit in FLMCR2, and finally setting the E bit. F initialized to H'80 by a reset, and in hardware standby mode, software standby mode, mode, subsleep mode, and watch mode. When on-chip flash memory is disabled, a rea H'00, and writes are invalid.

Writes to the EV and PV bits in FLMCR1 are enabled only when SWE=1; writes to th when SWE = 1, and ESU = 1; and writes to the P bit only when SWE = 1, and PSU =

Bit 7—Flash Write Enable (FWE): Controls programming and erasing of the on-chi memory. This bit cannot be modified and is always read as 1.

Bit 6—Software Write Enable (SWE): Enables or disables flash memory programm should be set before setting bits ESU, PSU, EV, PV, E, P, and EB9 to EB0, and shoul cleared at the same time as these bits.

Bit 6

SWE	Description
0	Writes disabled
1	Writes enabled

Bits 5 and 4—Reserved: These bits cannot be modified and are always read as 0.

	When SWE = 1

Bit 2—Program-Verify (PV): Selects program-verify mode transition or clearing. Do SWE, ESU, PSU, EV, E, or P bit at the same time.

Bit 2

PV	Description
0	Program-verify mode cleared
1	Transition to program-verify mode [Setting condition] When SWE = 1

Bit 1—Erase (E): Selects erase mode transition or clearing. Do not set the SWE, ESU, PV, or P bit at the same time.

Bit 1

E	Description
0	Erase mode cleared
1	Transition to erase mode [Setting condition] When SWE = 1, and ESU = 1

22.5.2 Flash Memory Control Register 2 (FLMCR2)

Bit	7	6	5	4	3	2	1
	FLER	—	—	—	—	—	ESU
Initial value	0	0	0	0	0	0	0
Read/Write	R	—	—	—	—	—	R/W

FLMCR2 is an 8-bit register that monitors the presence or absence of flash memory pr
protection (error protection) and performs setup for flash memory program/erase mod
is initialized to H'00 by a reset, and in hardware standby mode. The ESU and PSU bits
to 0 in software standby mode, subactive mode, subsleep mode, and watch mode.

When on-chip flash memory is disabled, a read will return H'00, and writes are invalid

Bit 7—Flash Memory Error (FLER): Indicates that an error has occurred during an
flash memory (programming or erasing). When FLER is set to 1, flash memory goes t
protection state.

	Flash memory program/erase protection (error protection) is enabled [Setting condition] See section 22.8.3, Error Protection

Bits 6 to 2—Reserved: Always write 0 when writing to these bits.

Bit 1—Erase Setup (ESU): Prepares for a transition to erase mode. Set this bit to 1 be
the E bit to 1 in FLMCR1. Do not set the SWE, PSU, EV, PV, E, or P bit at the same ti

Bit 1

ESU	Description
0	Erase setup cleared (Ini
1	Erase setup [Setting condition] When SWE = 1

Bit 0—Program Setup (PSU): Prepares for a transition to program mode. Set this bit t
setting the P bit to 1 in FLMCR1. Do not set the SWE, ESU, EV, PV, E, or P bit at the

Bit 0

PSU	Description
0	Program setup cleared (
1	Program setup [Setting condition] When SWE = 1

EBR2	EB7	EB6	EB5	EB4	EB3	EB2	EB1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W

Notes: 1. In normal mode, these bits cannot be modified and are always read as 0.
2. Bits EB8 and EB9 are not present in the 64-kbyte versions; they must not b

EBR1 and EBR2 are registers that specify the flash memory erase area block by block 2 in EBR1 (128 kB versions only) and bits 7 to 0 in EBR2 are readable/writable bits. I EBR2 are each initialized to H'00 by a reset, in hardware standby mode, software stan subactive mode, subsleep mode, and watch mode, and the SWE bit in FLMCR1 is not bit in EBR1 or EBR2 is set, the corresponding block can be erased. Other blocks are e protected. Set only one bit in EBR1 or EBR2 (more than one bit cannot be set). When flash memory is disabled, a read will return H'00, and writes are invalid.

The flash memory block configuration is shown in table 22.5.

EB4 (28 kbytes)	EB4 (28 kbytes)	H'(00)1000 to H'(00)7FFF
EB5 (16 kbytes)	EB5 (16 kbytes)	H'(00)8000 to H'(00)BFFF
EB6 (8 kbytes)	EB6 (8 kbytes)	H'(00)C000 to H'(00)DFFF
EB7 (8 kbytes)	EB7 (8 kbytes)	H'00E000 to H'00FFFF
EB8 (32 kbytes)	—	H'010000 to H'017FFF
EB9 (32 kbytes)	—	H'018000 to H'01FFFF

22.5.4 Serial/Timer Control Register (STCR)

Bit	7	6	5	4	3	2	1
	IICS	IICX1	IICX0	IICE	FLSHE	—	ICKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

STCR is an 8-bit readable/writable register that controls register access, the IIC operati (when the on-chip IIC option is included), and on-chip flash memory control (in F-ZTA versions), and also selects the TCNT input clock. For details on functions not related to flash memory, see section 3.2.4, Serial Timer Control Register (STCR), and description individual modules. If a module controlled by STCR is not used, do not write 1 to the corresponding bit.

STCR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 4—I^{2}C Control (IICS, IICX1, IICX0, IICE): When the on-chip IIC option i these bits control the operation of the I^{2}C bus interface. For details, see section 16, I^{2}C I Interface.

Bit 2—Reserved: Do not write 1 to this bit.

Bits 1 and 0—Internal Clock Select 1 and 0 (ICKS1, ICKS0): These bits control 8-
operation. See section 12, 8-Bit Timers, for details.

22.6 On-Board Programming Modes

When pins are set to on-board programming mode, program/erase/verify operations ca
performed on the on-chip flash memory. There are two on-board programming modes
and user program mode. The pin settings for transition to each of these modes are sho
22.6. For a diagram of the transitions to the various flash memory modes, see figure 2

Only advanced mode setting is possible for boot mode.

In the case of user program mode, established in advanced mode or normal mode, dep
the setting of the MD0 pin. In normal mode, only programming of a 56-kbyte area of
memory is possible.

Table 22.6 Setting On-Board Programming Modes

	Mode				
Mode Name	**CPU Operating Mode**	**MD1**	**MD0**	**P92**	**P91**
Boot mode	Advanced mode	0	0	1*	1*
User program mode	Advanced mode	1	0	—	—
	Normal mode		1	—	—

Note: * Can be used as I/O ports after boot mode is initiated.

the transfer is completed, control branches to the start address of the programming cont
program area and the programming control program execution state is entered (flash m
programming is performed).

The transferred programming control program must therefore include coding that follo
programming algorithm given later.

The system configuration in boot mode is shown in figure 22.7, and the boot program r
execution procedure in figure 22.8.

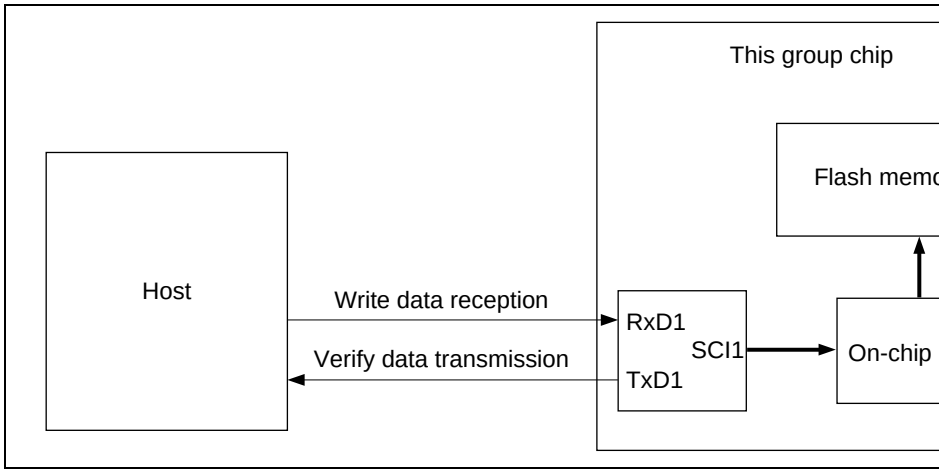
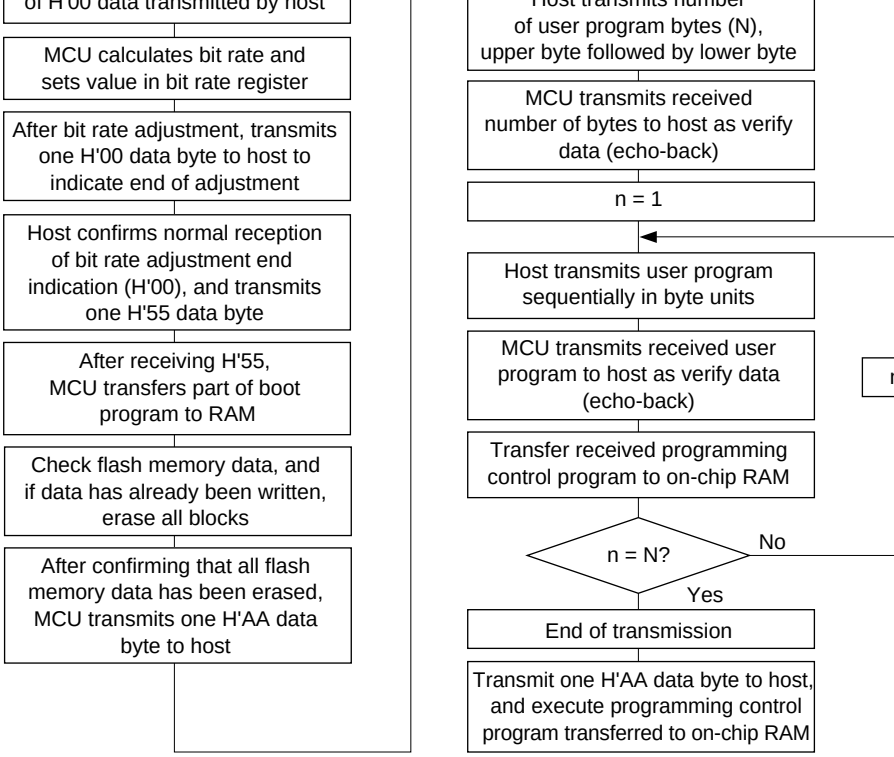


Figure 22.7 System Configuration in Boot Mode



Note: If a memory cell does not operate normally and cannot be erased, one H'FF b transmitted as an erase error, and the erase operation and subsequent opera are halted.

Figure 22.8 Boot Mode Execution Procedure

Figure 22.9 RxD1 Input Signal when Using Automatic SCI Bit Rate Adjust

When boot mode is initiated, this group MCU measures the low period of the asynchro communication data (H'00) transmitted continuously from the host. The SCI transmit/r format should be set as follows: 8-bit data, 1 stop bit, no parity. The MCU calculates th the transmission from the host from the measured low period, and transmits one H'00 b host to indicate the end of bit rate adjustment. The host should confirm that this adjustn indication (H'00) has been received normally, and transmit one H'55 byte to the MCU. cannot be performed normally, initiate boot mode again (reset), and repeat the above op Depending on the host's transmission bit rate and the MCU's system clock frequency, t be a discrepancy between the bit rates of the host and the MCU. To ensure correct SCI the host's transfer bit rate should be set to (2400, 4800, or 9600) bps.

Table 22.7 shows typical host transfer bit rates and system clock frequencies for which adjustment of the MCU's bit rate is possible. The boot program should be executed wit system clock range.

Table 22.7 System Clock Frequencies for which Automatic Adjustment of This C Rate Is Possible

Host Bit Rate	System Clock Frequency for which Automatic Adjustm of This Group Bit Rate Is Possible
9600 bps	8 MHz to 20 MHz
4800 bps	4 MHz to 20 MHz
2400 bps	2 MHz to 18 MHz

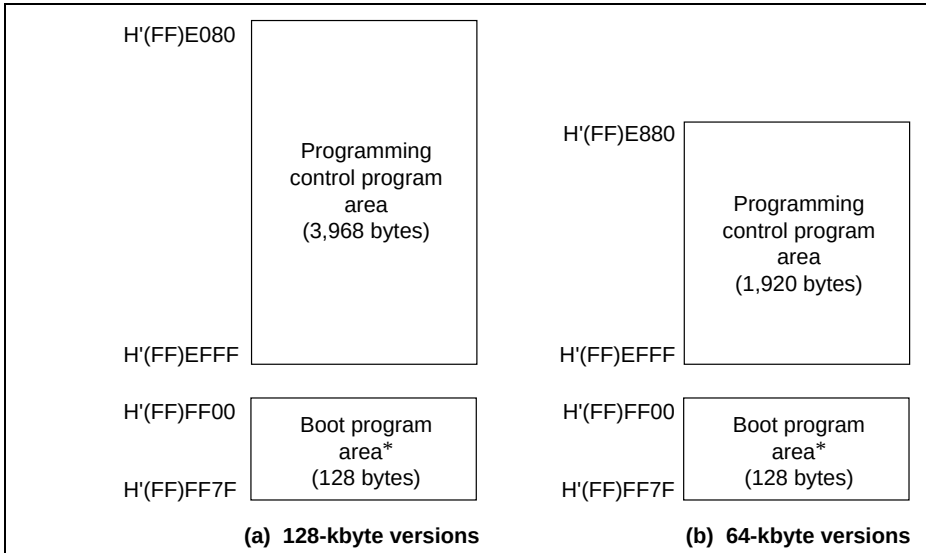


Figure 22.10 RAM Areas in Boot Mode

Notes on Use of User Mode

- When the chip comes out of reset in boot mode, it measures the low period of the i SCI's RxD1 pin. The reset should end with RxD1 high. After the reset ends, it take states for the chip to get ready to measure the low period of the RxD1 input.
- In boot mode, if any data has been programmed into the flash memory (if all data flash memory blocks are erased. Boot mode is for use when user program mode is such as the first time on-board programming is performed, or if the program activa program mode is accidentally erased.

The contents of the CPU's internal general registers are undefined at this time, so th registers must be initialized immediately after branching to the programming contro In particular, since the stack pointer (SP) is used implicitly in subroutine calls, etc., must be specified for use by the programming control program.

The initial values of other on-chip registers are not changed.

- Boot mode can be entered by making the pin settings shown in table 22.6 and execu reset-start.

  When the chip detects the boot mode setting at reset release[*1], P92, P91, and P90 ca as I/O ports.

  Boot mode can be cleared by driving the reset pin low, waiting at least 20 states, the the mode pins, and executing reset release[*1]. Boot mode can also be cleared by a W overflow reset.

  The mode pin input levels must not be changed in boot mode.

- If the mode pin input levels are changed (for example, from low to high) during a re state of ports with multiplexed address functions and bus control output pins ($\overline{AS}$, $\overline{R}$ will change according to the change in the microcomputer's operating mode[*2].

  Therefore, care must be taken to make pin settings to prevent these pins from becon signal pins during a reset, or to prevent collision with signals outside the microcomp

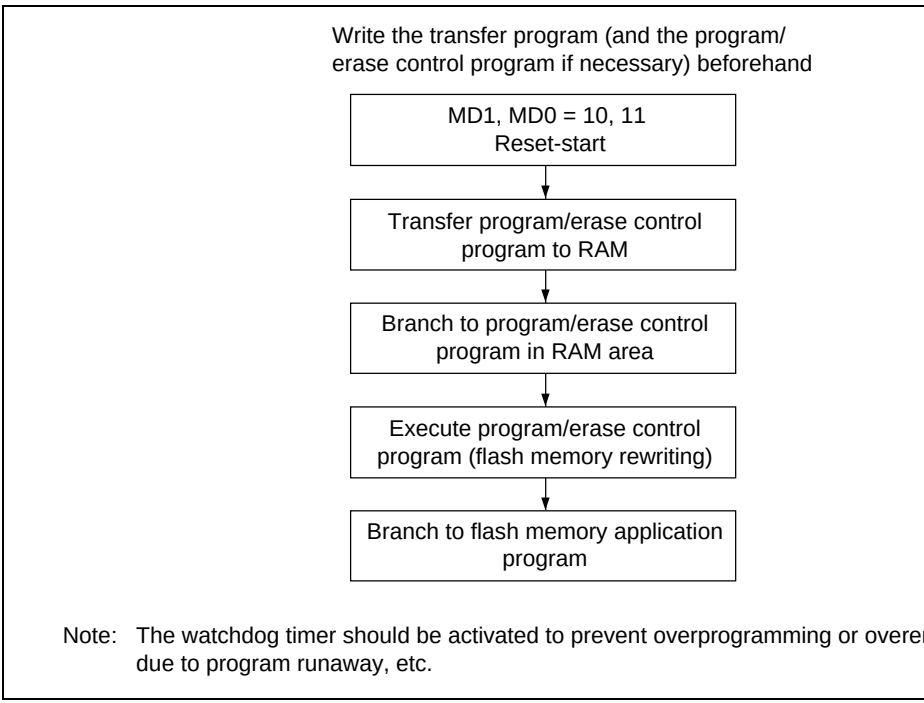
Notes: 1. Mode pin input must satisfy the mode programming setup time (t_{MDS} = 4 stat respect to the reset release timing.

2. Ports with multiplexed address functions will output a low level as the addre mode pin setting is for mode 1 is entered during a reset. In other modes, the go to the high-impedance state. The bus control output signals will output a if mode pin setting is for mode 1 is entered during a reset. In other modes, th go to the high-impedance state.

would in mode 2 and 3.

The flash memory itself cannot be read while the SWE bit is set to 1 to perform progr
erasing, so the control program that performs programming and erasing should be run
RAM or external memory.

Figure 22.11 shows the procedure for executing the program/erase control program wh
transferred to on-chip RAM.



Note: The watchdog timer should be activated to prevent overprogramming or overer
due to program runaway, etc.

Figure 22.11 User Program Mode Execution Procedure

located and executed in on-chip RAM or external memory.

Notes: 1. Operation is not guaranteed if setting/resetting of the SWE, EV, PV, E, and
FLMCR1, and the ESU and PSU bits in FLMCR2, is executed by a program
memory.
2. Perform programming in the erased state. Do not perform additional progra
previously programmed addresses.

22.7.1 Program Mode

Follow the procedure shown in the program/program-verify flowchart in figure 21.12 t
or programs to flash memory. Performing program operations according to this flowcha
enable data or programs to be written to flash memory without subjecting the device to
stress or sacrificing program data reliability. Programming should be carried out 32 byt
time.

For the wait times (x, y, z, α, β, γ, ε, η) after setting/clearing individual bits in flash me
control registers 1 and 2 (FLMCR1, FLMCR2) and the maximum number of writes (N
section 26.2.6, Flash Memory Characteristics.

Following the elapse of (x) µs or more after the SWE bit is set to 1 in flash memory co
register 1 (FLMCR1), 32-byte program data is stored in the program data area and repr
area, and the 32-byte data in the reprogram data area written consecutively to the write
The lower 8 bits of the first address written to must be H'00, H'20, H'40, H'60, H'80, H'
or H'E0. Thirty-two consecutive byte data transfers are performed. The program addres
program data are latched in the flash memory. A 32-byte data transfer must be perform
writing fewer than 32 bytes; in this case, H'FF data must be written to the extra address

Next, the watchdog timer is set to prevent overprogramming in the event of program ru
Set a value greater than (y + z + α + β) µs as the WDT overflow period. After this, prep
program mode (program setup) is carried out by setting the PSU bit in FLMCR2, and a
elapse of (y) µs or more, the operating mode is switched to program mode by setting th

After the elapse of a given programming time, the programming mode is exited (the P
FLMCR1 is cleared, then the PSU bit in FLMCR2 is cleared at least (α) µs later). The
timer is cleared after the elapse of (β) µs or more, and the operating mode is switched
verify mode by setting the PV bit in FLMCR1. Before reading in program-verify mod
write of H'FF data should be made to the addresses to be read. The dummy write shou
executed after the elapse of (γ) µs or more. When the flash memory is read in this stat
is read in 16-bit units), the data at the latched address is read. Wait at least (ε) µs after
write before performing this read operation. Next, the originally written data is compa
verify data, and reprogram data is computed (see figure 22.12) and transferred to the r
data area. After 32 bytes of data have been verified, exit program-verify mode, wait fo
µs, then clear the SWE bit in FLMCR1. If reprogramming is necessary, set program m
and repeat the program/program-verify sequence as before. However, ensure that the
program/program-verify sequence is not repeated more than (N) times on the same bit

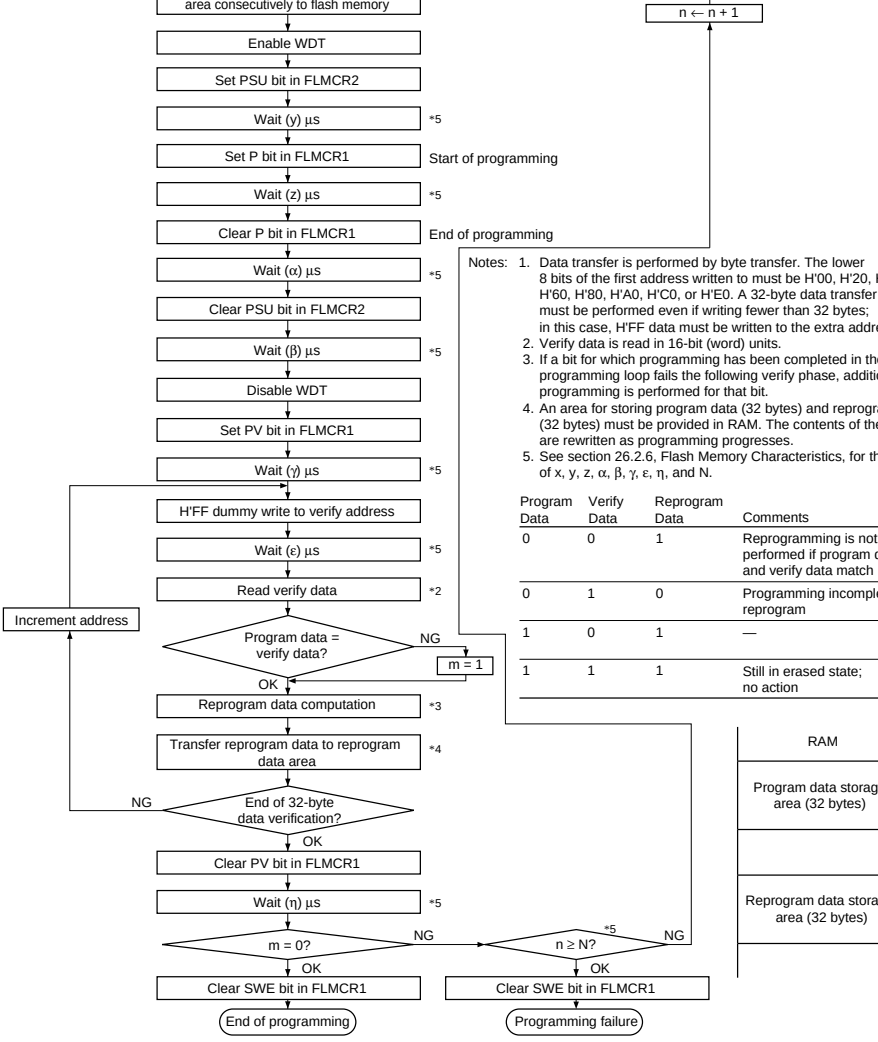


Figure 22.12 Program/Program-Verify Flowchart

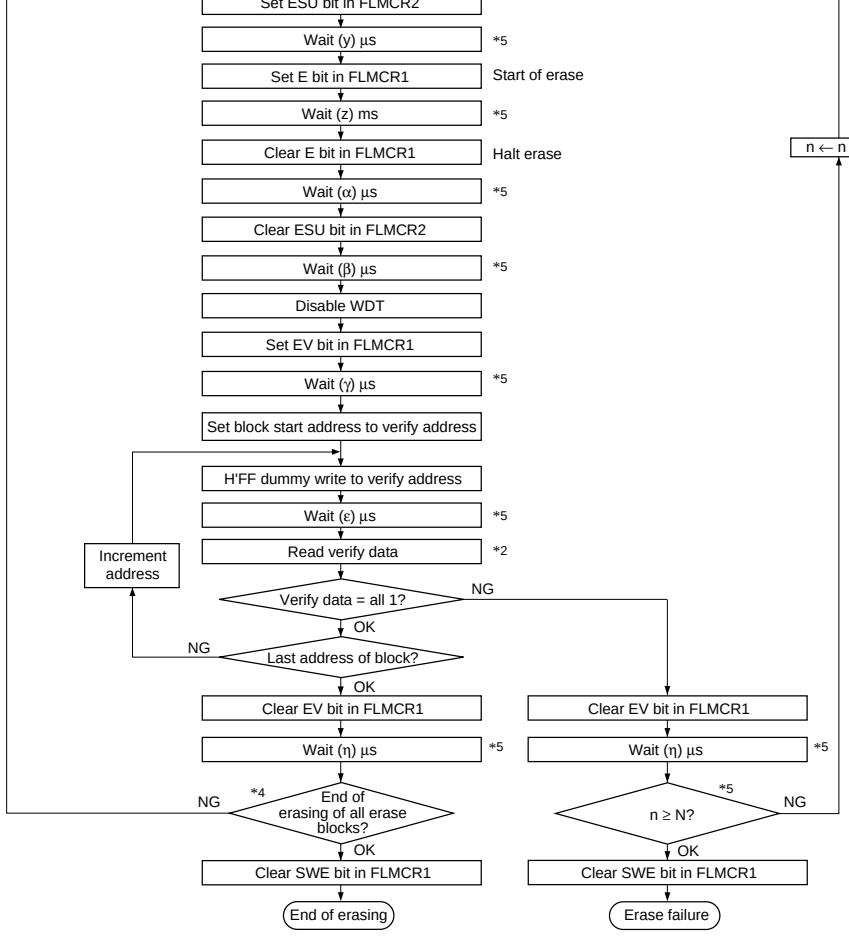
To perform data or program erasure, make a 1 bit setting for the flash memory area to erase block register 1 or 2 (EBR1 or EBR2) at least (x) µs after setting the SWE bit to memory control register 1 (FLMCR1). Next, the watchdog timer is set to prevent over the event of program runaway, etc. Set a value greater than (y + z + α + β) ms as the W overflow period. After this, preparation for erase mode (erase setup) is carried out by s ESU bit in FLMCR2, and after the elapse of (y) µs or more, the operating mode is swi erase mode by setting the E bit in FLMCR1. The time during which the E bit is set is t memory erase time. Ensure that the erase time does not exceed (z) ms.

Note: With flash memory erasing, preprogramming (setting all data in the memory t to 0) is not necessary before starting the erase procedure.

22.7.4 Erase-Verify Mode

In erase-verify mode, data is read after memory has been erased to check whether it ha correctly erased.

After the elapse of the erase time, erase mode is exited (the E bit in FLMCR1 is cleare ESU bit in FLMCR2 is cleared at least (α) µs later), the watchdog timer is cleared afte of (β) µs or more, and the operating mode is switched to erase-verify mode by setting FLMCR1. Before reading in erase-verify mode, a dummy write of H'FF data should b the addresses to be read. The dummy write should be executed after the elapse of (γ) µ When the flash memory is read in this state (verify data is read in 16-bit units), the dat latched address is read. Wait at least (ε) µs after the dummy write before performing t operation. If the read data has been erased (all 1), a dummy write is performed to the n and erase-verify is performed. If the read data has not been erased, set erase mode aga repeat the erase/erase-verify sequence in the same way. However, ensure that the erase verify sequence is not repeated more than (N) times. When verification is completed, verify mode, and wait for at least (η) µs. If erasure has been completed on all the erase clear the SWE bit in FLMCR1. If there are any unerased blocks, make a 1 bit setting i EBR2 for the flash memory area to be erased, and repeat the erase/erase-verify sequen same way.



Notes: 1. Preprogramming (setting erase block data to all 0) is not necessary.
2. Verify data is read in 16-bit (W) units.
3. Set only one bit in EBR1or EBR2. More than one bit cannot be set.
4. Erasing is performed in block units. To erase a number of blocks, the individual blocks must be erased sequentially.
5. See section 26.2.6, Flash Memory Characteristics, for the values of x, y, z, α, β, γ, ε, η, and N.

Figure 22.13 Erase/Erase-Verify Flowchart (Single-Block Erase)

disabled or aborted. Hardware protection is reset by settings in flash memory control r
and 2 (FLMCR1, FLMCR2) and erase block registers 1 and 2 (EBR1, EBR2). (See tab

Table 22.8 Hardware Protection

Item	Description	Func Program
Reset/standby protection	• In a reset (including a WDT overflow reset), software standby mode, subactive mode, subsleep mode, and watch mode, FLMCR1, FLMCR2, EBR1, and EBR2 are initialized, and the program/erase-protected state is entered. • 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, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width specified in the AC Characteristics section.	Yes

22.8.2 Software Protection

Software protection can be implemented by setting the SWE bit in FLMCR1 and erase
registers 1 and 2 (EBR1, EBR2). When software protection is in effect, setting the P o
flash memory control register 1 (FLMCR1) does not cause a transition to program mo
mode. (See table 22.9.)

protection	by settings in erase block registers 1 and 2 (EBR1, EBR2). • Setting EBR1 and EBR2 to H'00 places all blocks in the erase-protected state.

22.8.3 Error Protection

In error protection, an error is detected when MCU runaway occurs during flash memo programming/erasing, or operation is not performed in accordance with the program/er algorithm, and the program/erase operation is aborted. Aborting the program/erase oper prevents damage to the flash memory due to overprogramming or overerasing.

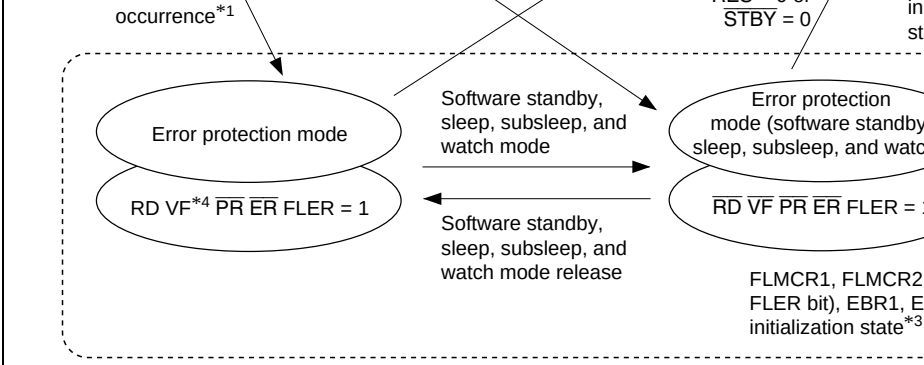
If the MCU malfunctions during flash memory programming/erasing, the FLER bit is s FLMCR2 and the error protection state is entered. The FLMCR1, FLMCR2, EBR1, an settings are retained, but program mode or erase mode is aborted at the point at which t occurred. Program mode or erase mode cannot be re-entered by re-setting the P or E bi PV and EV bit setting is enabled, and a transition can be made to verify mode.

FLER bit setting conditions are as follows:

- When flash memory is read during programming/erasing (including a vector read o fetch)
- Immediately after exception handling (excluding a reset) during programming/erasi
- When a SLEEP instruction (transition to software standby, sleep, subactive, subslee mode) is executed during programming/erasing
- When the bus is released during programming/erasing

Error protection is released only by a reset and in hardware standby mode.

Figure 22.14 shows the flash memory state transition diagram.



Legend:

RD: Memory read possible	$\overline{RD}$: Memory read not possible
VF: Verify-read possible	$\overline{VF}$: Verify-read not possible
PR: Programming possible	$\overline{PR}$: Programming not possible
ER: Erasing possible	$\overline{ER}$: Erasing not possible

Notes: 1. When an error occurs other than due to a SLEEP instruction, or when a SLEEP instruc executed for a transition to subactive mode
2. When an error occurs due to a SLEEP instruction (except subactive mode)
3. Except sleep mode
4. $\overline{VF}$ in subactive mode

Figure 22.14 Flash Memory State Transitions

2. In the interrupt exception handling sequence during programming or erasing, the ve not be read correctly*[2], possibly resulting in MCU runaway.
3. If interrupt occurred during boot program execution, it would not be possible to exe normal boot mode sequence.

For these reasons, in on-board programming mode alone there are conditions for disabl interrupt, as an exception to the general rule. However, this provision does not guarante erasing and programming or MCU operation. All interrupt requests, including NMI inp therefore be disabled inside and outside the MCU when programming or erasing flash m Interrupt is also disabled in the error-protection state while the P or E bit remains set in

Notes: 1. Interrupt requests must be disabled inside and outside the MCU until the pro control program has completed programming.
2. The vector may not be read correctly in this case for the following two reaso
   - If flash memory is read while being programmed or erased (while the P set in FLMCR1), correct read data will not be obtained (undetermined va be returned).
   - If the interrupt entry in the vector table has not been programmed yet, in exception handling will not be executed correctly.

auto-erase mode, and status read mode are supported with these device types. In auto-
mode, auto-erase mode, and status read mode, a status polling procedure is used, and i
mode, detailed internal signals are output after execution of an auto-program or auto-e
operation.

Table 22.10 shows programmer mode pin settings.

Notes: 1. Applies to the H8S/2148 and H8S/2144.
2. Applies to the H8/2147N and H8S/2142.
3. Use products other than the A-mask version of the H8S/2148, H8S/2147N,
and H8S/2142 (in either 5-V or 3-V version) with the writing voltage for th
programmer set to 5.0 V. Do not use the A-mask version with a 5.0-V PRO
programmer setting.

Table 22.10 Programmer Mode Pin Settings

Pin Names	Setting/External Circuit Connectio
Mode pins: MD1, MD0	Low-level input to MD1, MD0
$\overline{\text{STBY}}$ pin	High-level input (Hardware standby mode not set)
$\overline{\text{RES}}$ pin	Power-on reset circuit
XTAL and EXTAL pins	Oscillation circuit
Other setting pins: P97, P92, P91, P90, P67	Low-level input to P92, P67, high-level input to P97, P91, P90

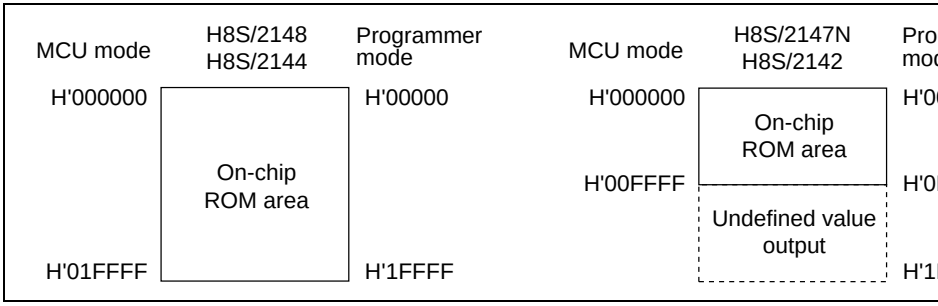


Figure 22.15 Memory Map in Programmer mode

22.10.3 Programmer Mode Operation

Table 22.11 shows how the different operating modes are set when using programmer m
table 22.12 lists the commands used in programmer mode. Details of each mode are giv

- Memory Read Mode

  Memory read mode supports byte reads.
- Auto-Program Mode

  Auto-program mode supports programming of 128 bytes at a time. Status polling is
  confirm the end of auto-programming.
- Auto-Erase Mode

  Auto-erase mode supports automatic erasing of the entire flash memory. Status poll
  to confirm the end of auto-erasing.
- Status Read Mode

  Status polling is used for auto-programming and auto-erasing, and normal terminati
  confirmed by reading the FO6 signal. In status read mode, error information is outp
  error occurs.

Notes: 1. Chip disable is not a standby state; internally, it is an operation state.
2. Ain indicates that there is also address input in auto-program mode.

Table 22.12 Programmer Mode Commands

Command Name	Number of Cycles	1st Cycle			2nd Cy	
		Mode	Address	Data	Mode	Addres
Memory read mode	1 + n	Write	X	H'00	Read	RA
Auto-program mode	129	Write	X	H'40	Write	WA
Auto-erase mode	2	Write	X	H'20	Write	X
Status read mode	2	Write	X	H'71	Write	X

Notes: 1. In auto-program mode, 129 cycles are required for command writing by a s
128-byte write.
2. In memory read mode, the number of cycles depends on the number of ad
cycles (n).

22.10.4 Memory Read Mode

- After the end of an auto-program, auto-erase, or status read operation, the comma
is entered. To read memory contents, a transition must be made to memory read m
means of a command write before the read is executed.
- Command writes can be performed in memory read mode, just as in the command
- Once memory read mode has been entered, consecutive reads can be performed.
- After power-on, memory read mode is entered.

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

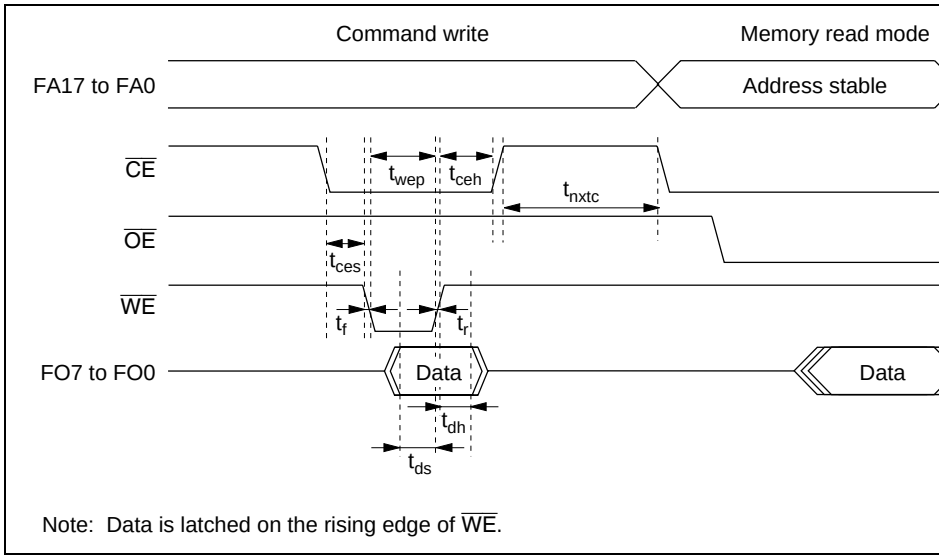


Figure 22.16 Memory Read Mode Timing Waveforms after Command W

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

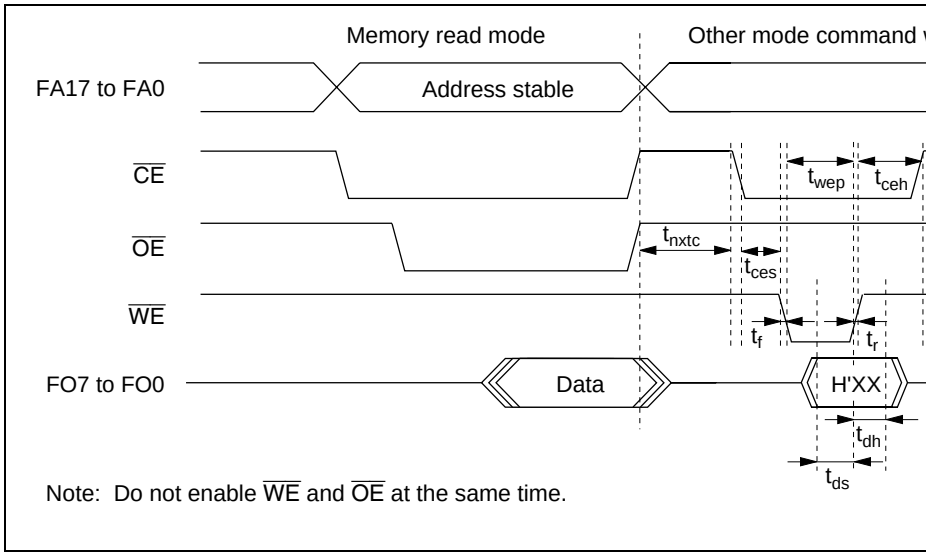


Figure 22.17 Timing Waveforms when Entering Another Mode from Memory

Output disable delay time	t_{df}	—	100	ns
Data output hold time	t_{oh}	5	—	ns

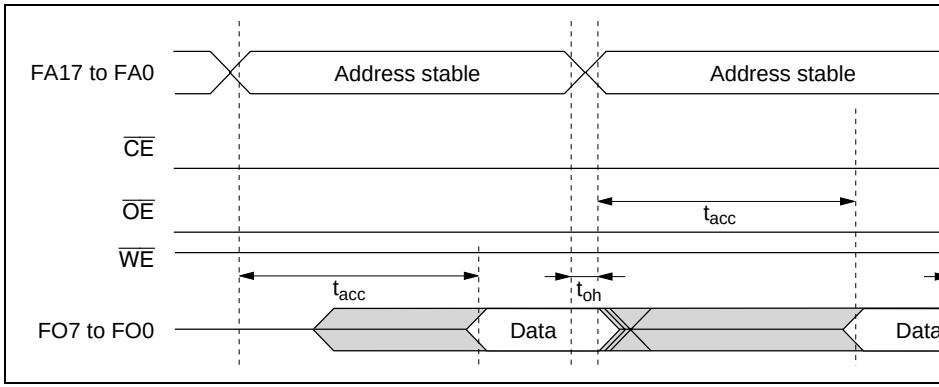


Figure 22.18 Timing Waveforms for $\overline{CE}$/$\overline{OE}$ Enable State Read

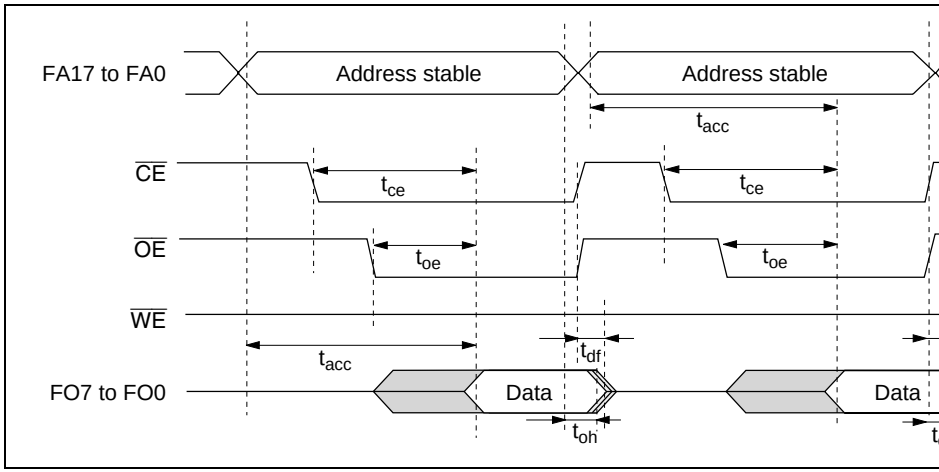


Figure 22.19 Timing Waveforms for $\overline{CE}$/$\overline{OE}$ Clocked Read

Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
Status polling start time	t_{wsts}	1	—	ms
Status polling access time	t_{spa}	—	150	ns
Address setup time	t_{as}	0	—	ns
Address hold time	t_{ah}	60	—	ns
Memory write time	t_{write}	1	3000	ms
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

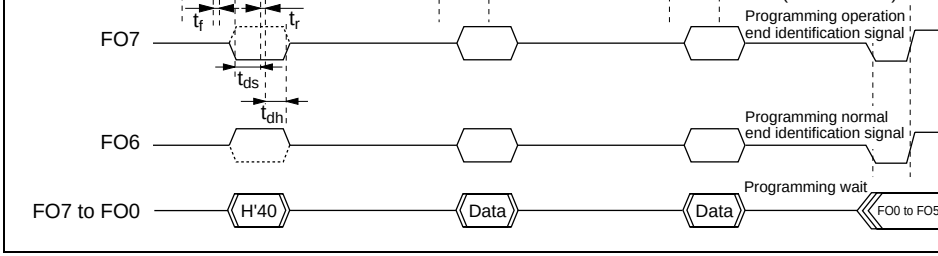


Figure 22.20 Auto-Program Mode Timing Waveforms

Notes on Use of Auto-Program Mode

- In auto-program mode, 128 bytes are programmed simultaneously. This should be c by executing 128 consecutive byte transfers.
- A 128-byte data transfer is necessary even when programming fewer than 128 bytes case, H'FF data must be written to the extra addresses.
- The lower 8 bits of the transfer address must be H'00 or H'80. If a value other than a address is input, processing will switch to a memory write operation but a write erro flagged.
- Memory address transfer is performed in the second cycle (figure 22.20). Do not pe transfer after the second cycle.
- Do not perform a command write during a programming operation.
- Perform one auto-programming operation for a 128-byte block for each address. Characteristics are not guaranteed for two or more programming operations.
- Confirm normal end of auto-programming by checking FO6. Alternatively, status re can also be used for this purpose (FO7 status polling uses the auto-program operatio identification pin).
- The status polling FO6 and FO7 pin information is retained until the next command Until the next command write is performed, reading is possible by enabling $\overline{CE}$ and

Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
Status polling start time	t_{ests}	1	—	ms
Status polling access time	t_{spa}	—	150	ns
Memory erase time	t_{erase}	100	40000	ms
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

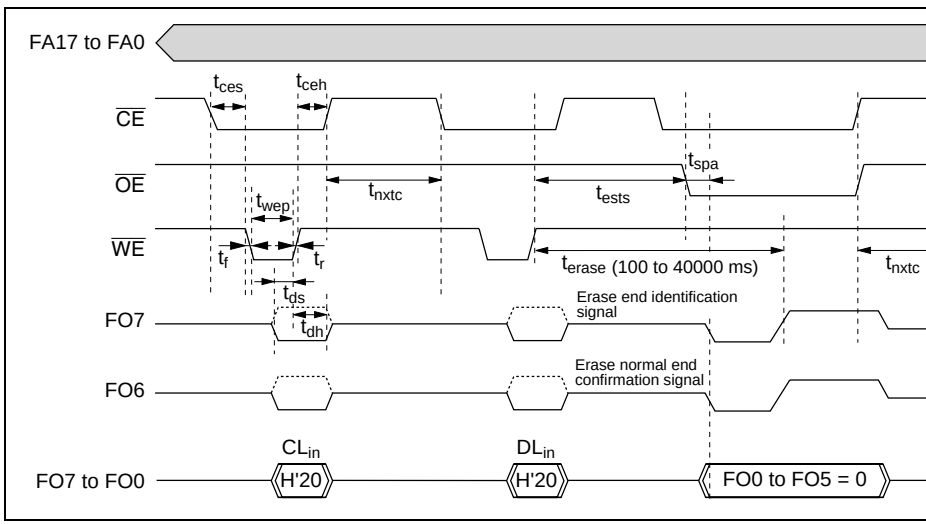


Figure 22.21 Auto-Erase Mode Timing Waveforms

Until the next command write is performed, reading is possible by enabling CE and

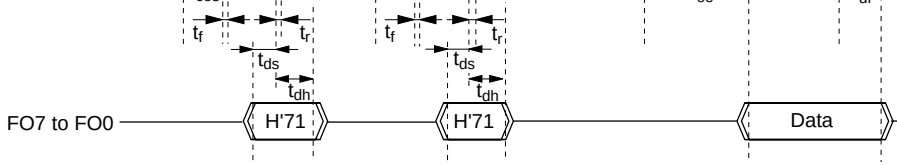
22.10.7 Status Read Mode

- Status read mode is used to identify what type of abnormal end has occurred. Use th when an abnormal end occurs in auto-program mode or auto-erase mode.
- The return code is retained until a command write for other than status read mode is performed.

Table 22.18 AC Characteristics in Status Read Mode

Conditions: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, T_a = 25°C ±5°C

Item	Symbol	Min	Max	Unit
Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{OE}$ output delay time	t_{oe}	—	150	ns
Disable delay time	t_{df}	—	100	ns
$\overline{CE}$ output delay time	t_{ce}	—	150	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns



Note: FO2 and FO3 are undefined.

Figure 22.22 Status Read Mode Timing Waveforms

Table 22.19 Status Read Mode Return Commands

Pin Name	FO7	FO6	FO5	FO4	FO3	FO2	FO1
Attribute	Normal end identification	Command error	Program-ming error	Erase error	—	—	Program-ming or erase count exceeded
Initial value	0	0	0	0	0	0	0
Indications	Normal end: 0 Abnormal end: 1	Command error: 1 Otherwise: 0	Program-ming error: 1 Otherwise: 0	Erase error: 1 Otherwise: 0	—	—	Count exceeded: 1 Otherwise: 0

Note: FO2 and FO3 are undefined.

22.10.8 Status Polling

- The FO7 status polling flag indicates the operating status in auto-program or auto-
- The FO6 status polling flag indicates a normal or abnormal end in auto-program o mode.

22.10.9 Programmer Mode Transition Time

Commands cannot be accepted during the oscillation stabilization period or the program
setup period. After the programmer mode setup time, a transition is made to memory re

Table 22.21 Command Wait State Transition Time Specifications

Item	Symbol	Min	Max
Standby release (oscillation stabilization time)	t_{osc1}	20	—
Programmer mode setup time	t_{bmv}	10	—
V_{CC} hold time	t_{dwn}	0	—

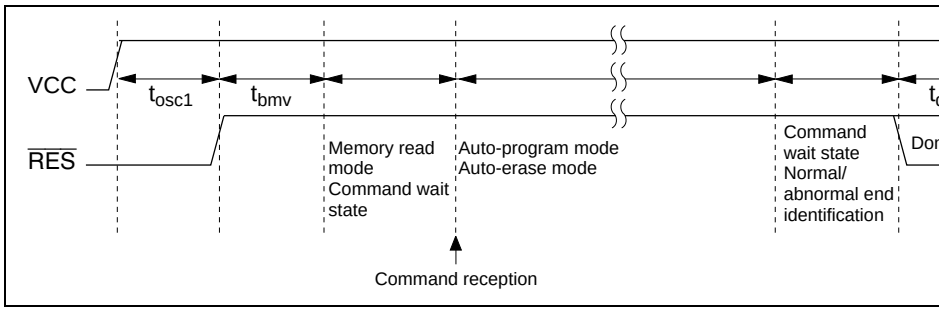


**Figure 22.23 Oscillation Stabilization Time, Programmer Mode Setup Time, ar
Supply Fall Sequence**

Notes: 1. The flash memory is initially in the erased state when the device is shippe
For other chips for which the erasure history is unknown, it is recommende
erasing be executed to check and supplement the initialization (erase) level
2. Auto-programming should be performed once only on the same address bl

22.11 Flash Memory Programming and Erasing Precautions

Precautions concerning the use of on-board programming mode and programmer mod
summarized below.

Use the specified voltages and timing for programming and erasing.
Applied voltages in excess of the rating can permanently damage the device. For a PR
programmer, use Renesas Technology microcomputer device types with 128-kbyte or
on-chip flash memory that support a 5.0-V programming voltage.
Do not select the HN28F101 or use a programming voltage of 3.3 V for the PROM pr
and only use the specified socket adapter. Incorrect use will result in damaging the de

Powering on and off
When applying or disconnecting V_{CC}, fix the $\overline{RES}$ pin low and place the flash memory
hardware protection state.
The power-on and power-off timing requirements should also be satisfied in the event
failure and subsequent recovery.

Use the recommended algorithm when programming and erasing flash memory.
The recommended algorithm enables programming and erasing to be carried out witho
subjecting the device to voltage stress or sacrificing program data reliability. When se
E bit in FLMCR1, the watchdog timer should be set beforehand as a precaution agains
runaway, etc.

Do not set or clear the SWE bit during program execution in flash memory.
Clear the SWE bit before executing a program or reading data in flash memory. When
bit is set, data in flash memory can be rewritten, but flash memory should only be acc
verify operations (verification during programming/erasing).

block erased.

Before programming, check that the chip is correctly mounted in the PROM prog
Overcurrent damage to the device can result if the index marks on the PROM programr socket adapter, and chip are not correctly aligned.

Do not touch the socket adapter or chip during programming.
Touching either of these can cause contact faults and write errors.

22.12 Note on Switching from F-ZTAT Version to Mask ROM Ver

The mask ROM version does not have the internal registers for flash memory control th provided in the F-ZTAT version. Table 22.22 lists the registers that are present in the F version but not in the mask ROM version. If a register listed in table 22.22 is read in th ROM version, an undefined value will be returned, Therefore, if application software d on the F-ZTAT version is switched to a mask ROM version product, it must be modifie that the registers in table 22.22 have no effect.

Table 22.22 Registers Present in F-ZTAT Version but Absent in Mask ROM Vers

Register	Abbreviation	Address
Flash memory control register 1	FLMCR1	H'FF80
Flash memory control register 2	FLMCR2	H'FF81
Erase block register 1	EBR1	H'FF82
Erase block register 2	EBR2	H'FF83

H8S/2148 F-ZTAT A-mask version and H8S/2144 F-ZTAT A-mask version have 128
H8S/2147 F-ZTAT A-mask version has 64 kbytes of on-chip flash memory. The flash
connected to the bus master by a 16-bit data bus. The bus master accesses both byte an
in one state, enabling faster instruction fetches and higher processing speed.

The mode pins (MD1 and MD0) and the EXPE bit in MDCR can be set to enable or d
on-chip ROM.

The flash memory versions of this group can be erased and programmed on-board as w
a general-purpose PROM programmer.

23.1.1 Block Diagram

Figure 23.1 shows a block diagram of the ROM.

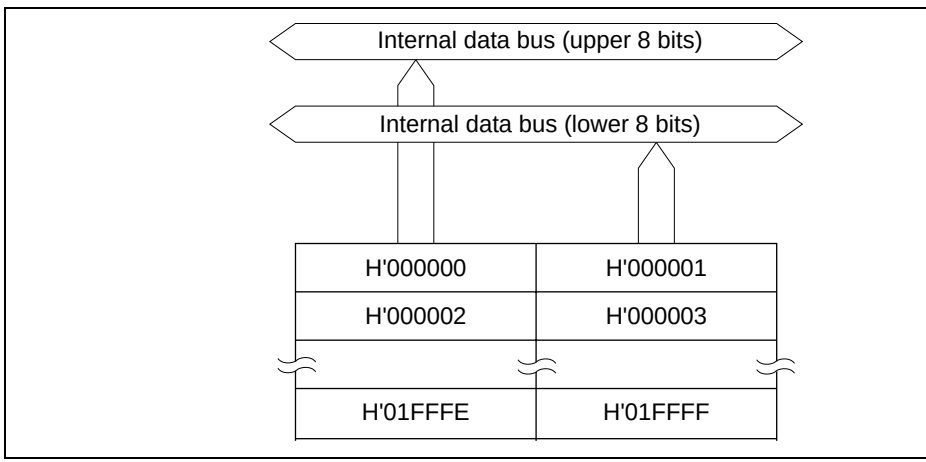


Figure 23.1 ROM Block Diagram
(A-mask versions of the H8S/2148 F-ZTAT and H8S/2144 F-ZTAT)

Note: * Lower 16 bits of the address.

23.2 Register Descriptions

23.2.1 Mode Control Register (MDCR)

Bit	7	6	5	4	3	2	1
	EXPE	—	—	—	—	—	MDS1
Initial value	—*	0	0	0	0	0	—*
Read/Write	R/W*	—	—	—	—	—	R

Note: * Determined by the MD1 and MD0 pins.

MDCR is an 8-bit read-only register used to set this group operating mode and monitor operating mode.

The EXPE bit is initialized in accordance with the mode pin states by a reset and in hardstandby mode.

Bit 7—Expanded Mode Enable (EXPE): Sets expanded mode. In mode 1, EXPE is fi and cannot be modified. In modes 2 and 3, EXPE has an initial value of 0 and can be re written.

Bit 7

EXPE	Description
0	Single-chip mode selected
1	Expanded mode selected

The on-chip flash memory is connected to the CPU by a 16-bit data bus, and both byte data is accessed in one state. Even addresses are connected to the upper 8 bits, and odd to the lower 8 bits. Word data must start at an even address.

The mode pins (MD1 and MD0) and the EXPE bit in MDCR can be set to enable or d on-chip ROM, as shown in table 23.2.

In normal mode, the maximum amount of ROM that can be used is 56 kbytes.

Table 23.2 Operating Modes and ROM

Operating Mode						
MCU Operating Mode	**CPU Operating Mode**	**Description**	**Mode Pins**		**MDCR**	
			MD1	**MD0**	**EXPE**	**On-**
Mode 1	Normal	Expanded mode with on-chip ROM disabled	0	1	1	Disa
Mode 2	Advanced	Single-chip mode	1	0	0	Ena
	Advanced	Expanded mode with on-chip ROM enabled			1	
Mode 3	Normal	Single-chip mode		1	0	Ena
	Normal	Expanded mode with on-chip ROM enabled			1	(56

Note: * H8S/2148 F-ZTAT A-mask version and H8S/2144 F-ZTAT A-mask version 128 kbytes, and H8S/2147 F-ZTAT A-mask version has 64 kbytes of on-ch

— Erase mode
— Program-verify mode
— Erase-verify mode

- Programming/erase methods

  The flash memory is programmed 128 bytes at a time. Erasing is performed by bloc
  single-block units). When erasing multiple blocks, the individual blocks must be er
  sequentially. Block erasing can be performed as required on 1-kbyte, 28-kbyte, 16-k
  kbyte, and 32-kbyte blocks.
- Programming/erase times

  The flash memory programming time is 10 ms (typ.) for simultaneous 128-byte pro
  equivalent to approximately 80 µs (typ.) per byte, and the erase time is 100 ms (typ
- Reprogramming capability

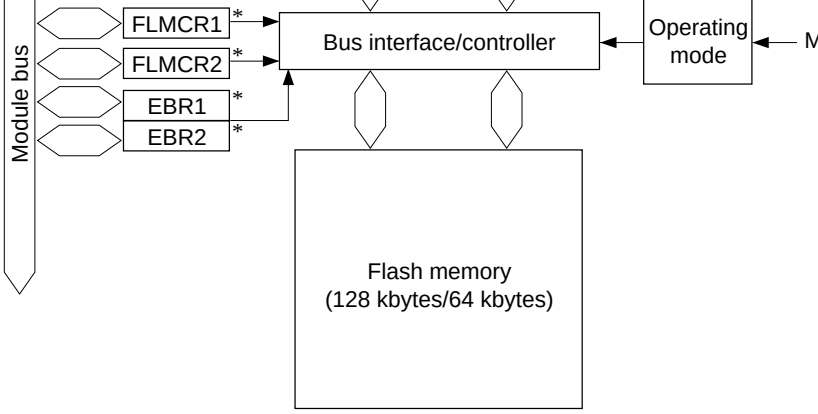
  The flash memory can be reprogrammed up to 100 times.
- On-board programming modes

  There are two modes in which flash memory can be programmed/erased/verified on
  — Boot mode
  — User program mode
- Automatic bit rate adjustment

  With data transfer in boot mode, the bit rate of the chip can be automatically adjuste
  the transfer bit rate of the host.
- Protect modes

  There are three protect modes, hardware, software, and error protect, which allow p
  status to be designated for flash memory program/erase/verify operations.
- Programmer mode

  Flash memory can be programmed/erased in programmer mode, using a PROM pro
  as well as in on-board programming mode.

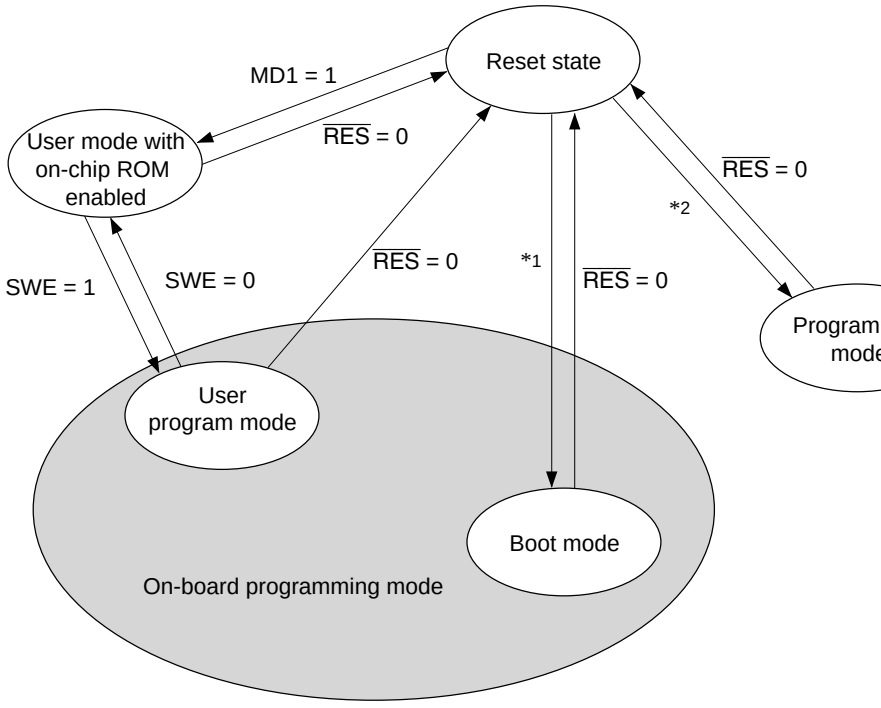


Legend:
FLMCR1: Flash memory control register 1
FLMCR2: Flash memory control register 2
EBR1: Erase block register 1
EBR2: Erase block register 2

Note: * These registers are used only in the flash memory version. In the mask ROM v
a read at any of these addresses will return an undefined value, and writes are

Figure 23.2 Block Diagram of Flash Memory

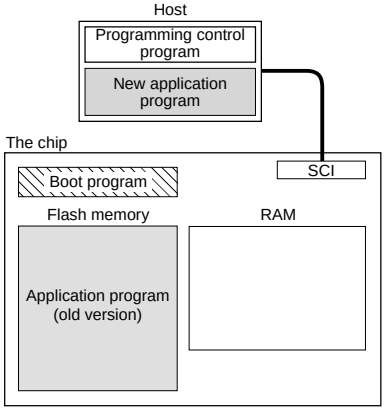
mode.

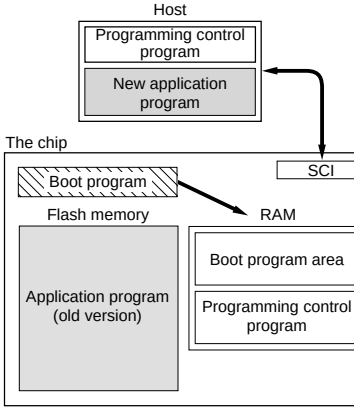


Notes: Only make a transition between user mode and user program mode when the C not accessing the flash memory.

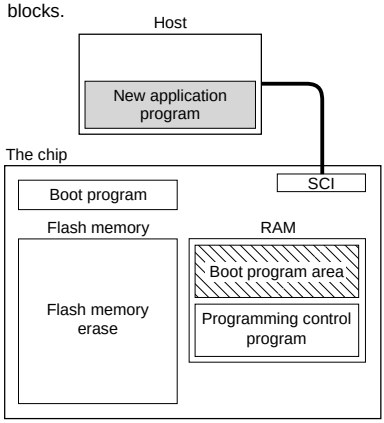
1. MD0 = MD1 = 0, P92 = P91 = P90 = 1
2. MD0 = MD1 = 0, P92 = 0, P91 = P90 = 1

Figure 23.3 Flash Memory Mode Transitions





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, entire flash memory erasure is performed, without regard to blocks.



4. Writing new application program
The programming control program transfe
from the host to RAM by SCI communicat
executed, and the new application progra
host is written into the flash memory.

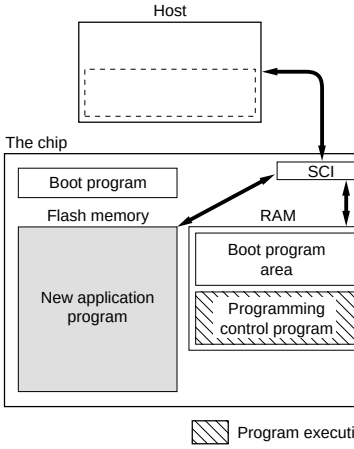


Figure 23.4 Boot Mode

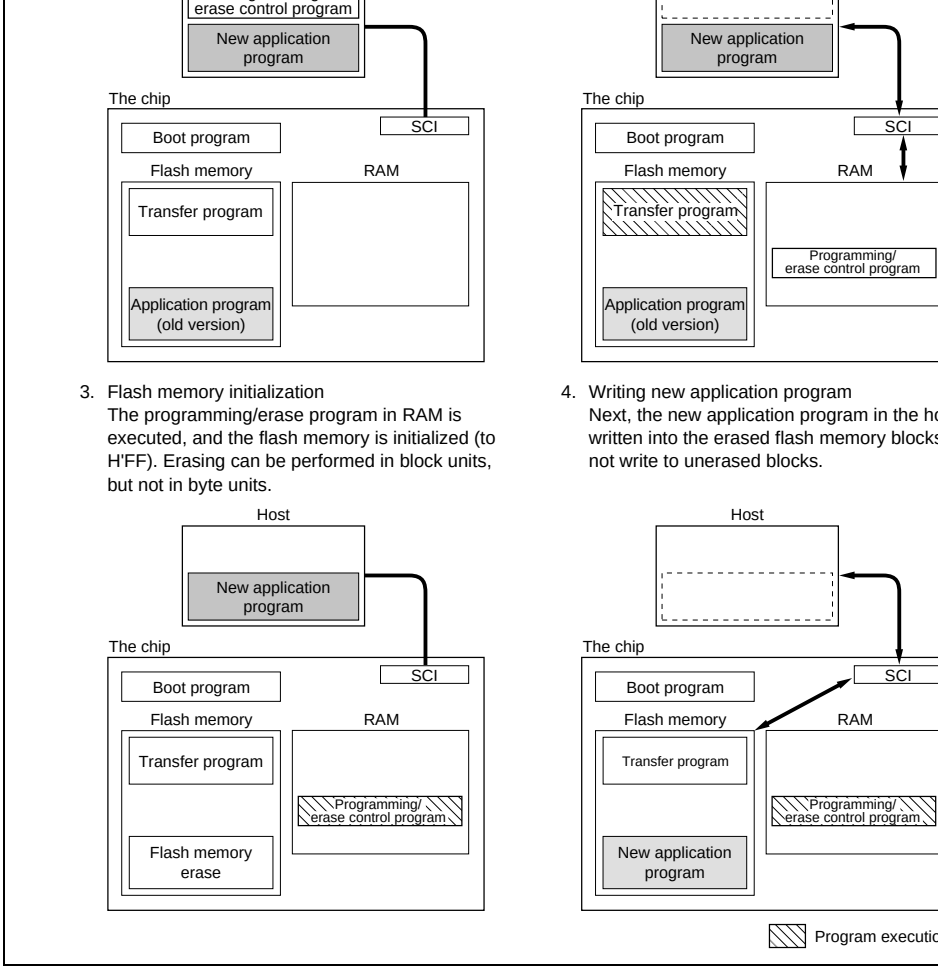


Figure 23.5 User Program Mode (Example)

Block Configuration

The flash memory is divided into two 32-kbyte blocks (128-kbyte version only), two 8
blocks, one 16-kbyte block, one 28-kbyte block, and four 1-kbyte blocks.

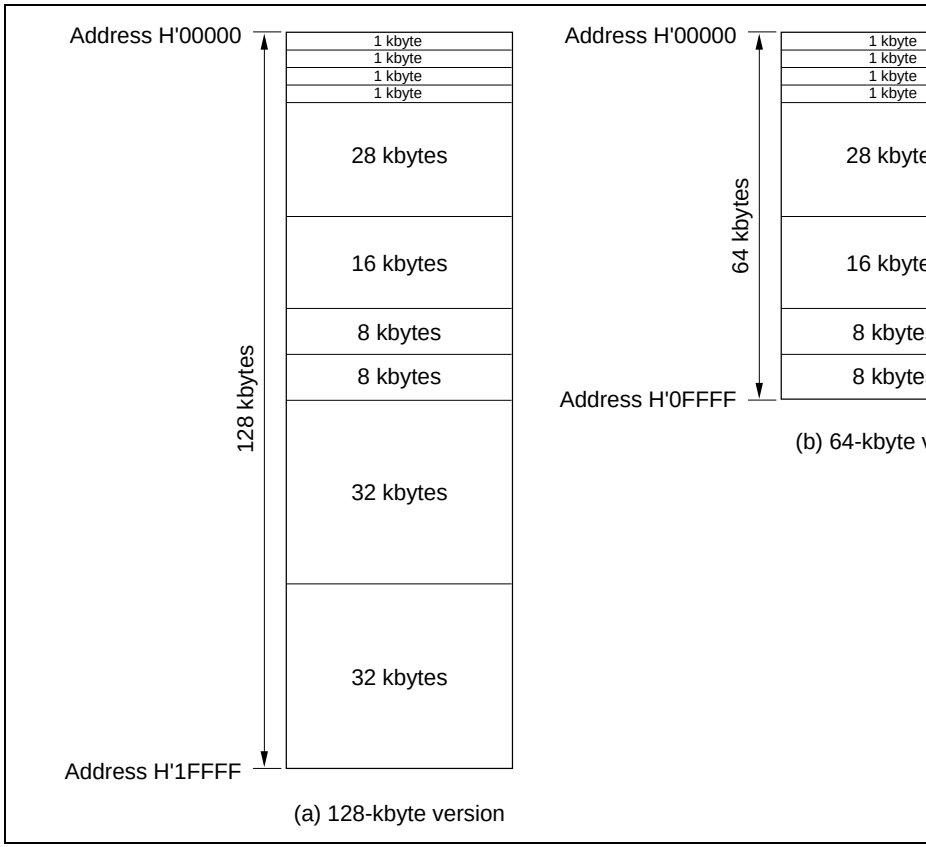


Figure 23.6 Flash Memory Block Configuration

Mode 0	MD0	Input	Sets MCU operating mode
Port 92	P92	Input	Sets MCU operating mode when MD1 =
Port 91	P91	Input	Sets MCU operating mode when MD1 =
Port 90	P90	Input	Sets MCU operating mode when MD1 =
Transmit data	TxD1	Output	Serial transmit data output
Receive data	RxD1	Input	Serial receive data input

23.4.5 Register Configuration

The registers used to control the on-chip flash memory when enabled are shown in tabl
In order for these registers to be accessed, the FLSHE bit must be set to 1 in STCR.

Table 23.4 Flash Memory Registers

Register Name	Abbreviation	R/W	Initial Value	Ad
Flash memory control register 1	FLMCR1*5	R/W*3	H'80	H'F
Flash memory control register 2	FLMCR2*5	R/W*3	H'00*4	H'F
Erase block register 1	EBR1*5	R/W*3	H'00*4	H'F
Erase block register 2	EBR2*5	R/W*3	H'00*4	H'F
Serial/timer control register	STCR	R/W	H'00	H'F

Notes: 1. Lower 16 bits of the address.
2. Flash memory registers share addresses with other registers. Register selec performed by the FLSHE bit in the serial/timer control register (STCR).
3. In modes in which the on-chip flash memory is disabled, a read will return H' writes are invalid.
4. When the SWE bit in FLMCR1 is not set, these registers are initialized to H'0
5. FLMCR1, FLMCR2, EBR1, and EBR2 are 8-bit registers. Only byte accesse for these registers, the access requiring 2 states.

Read/Write	R	R/W	R/W	R/W	R/W	R/W

FLMCR1 is an 8-bit register used for flash memory operating mode control. Program- or erase-verify mode is entered by setting SWE to 1. Program mode is entered by setti 1, then setting the PSU bit in FLMCR2, and finally setting the P bit. Erase mode is en setting SWE to 1, then setting the ESU bit in FLMCR2, and finally setting the E bit. F initialized to H'80 by a reset, and in hardware standby mode, software standby mode, mode, subsleep mode, and watch mode. When on-chip flash memory is disabled, a rea H'00, and writes are invalid.

Writes to the EV and PV bits in FLMCR1 are enabled only when SWE=1; writes to th when SWE = 1, and ESU = 1; and writes to the P bit only when SWE = 1, and PSU =

Bit 7—Flash Write Enable (FWE): Sets hardware protection against flash memory programming/erasing. This bit cannot be modified and is always read as 1.

Bit 6—Software Write Enable (SWE): Enables or disables flash memory programm should be set before setting bits ESU, PSU, EV, PV, E, P, and EB9 to EB0, and shoul cleared at the same time as these bits.

Bit 6

SWE	Description
0	Writes disabled
1	Writes enabled

Bit 5 and 4—Reserved: These bits cannot be modified and are always read as 0.

	When SWE = 1

Bit 2—Program-Verify (PV): Selects program-verify mode transition or clearing. Do SWE, ESU, PSU, EV, E, or P bit at the same time.

Bit 2

PV	Description
0	Program-verify mode cleared
1	Transition to program-verify mode [Setting condition] When SWE = 1

Bit 1—Erase (E): Selects erase mode transition or clearing. Do not set the SWE, ESU, PV, or P bit at the same time.

Bit 1

E	Description
0	Erase mode cleared
1	Transition to erase mode [Setting condition] When SWE = 1, and ESU = 1

23.5.2 Flash Memory Control Register 2 (FLMCR2)

Bit	7	6	5	4	3	2	1
	FLER	—	—	—	—	—	ESU
Initial value	0	0	0	0	0	0	0
Read/Write	R	—	—	—	—	—	R/W

FLMCR2 is an 8-bit register that monitors the presence or absence of flash memory p
protection (error protection) and performs setup for flash memory program/erase mod
is initialized to H'00 by a reset, and in hardware standby mode. The ESU and PSU bit
to 0 in software standby mode, subactive mode, subsleep mode, and watch mode.

When on-chip flash memory is disabled, a read will return H'00 and writes are invalid

Bit 7—Flash Memory Error (FLER): Indicates that an error has occurred during an
flash memory (programming or erasing). When FLER is set to 1, flash memory goes t
protection state.

	Flash memory program/erase protection (error protection) is enabled [Setting condition] See section 23.8.3, Error Protection

Bits 6 to 2—Reserved: Always write 0 when writing to these bits.

Bit 1—Erase Setup (ESU): Prepares for a transition to erase mode. Set this bit to 1 be
the E bit to 1 in FLMCR1. Do not set the SWE, PSU, EV, PV, E, or P bit at the same ti

Bit 1

ESU	Description
0	Erase setup cleared
1	Erase setup [Setting condition] When SWE = 1

Bit 0—Program Setup (PSU): Prepares for a transition to program mode. Set this bit t
setting the P bit to 1 in FLMCR1. Do not set the SWE, ESU, EV, PV, E, or P bit at the

Bit 0

PSU	Description
0	Program setup cleared
1	Program setup [Setting condition] When SWE = 1

EBR2	EB7	EB6	EB5	EB4	EB3	EB2	EB1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W*[1]	R/W	R/W	R/W	R/W	R/W	R/W

Notes: 1. In normal mode, these bits cannot be modified and are always read as 0.
2. Bits EB8 and EB9 are not present in the 64-kbyte versions; they must not b

EBR1 and EBR2 are registers that specify the flash memory erase area block by block 0 in EBR1 and bits 7 to 0 in EBR2 are readable/writable bits. EBR1 and EBR2 are eac to H'00 by a reset, in hardware standby mode, software standby mode, subactive mode mode, and watch mode, and when the SWE bit in FLMCR1 is not set. When a bit in E EBR2 is set, the corresponding block can be erased. Other blocks are erase-protected. bit in EBR1 and EBR2 (more than one bit cannot be set). When on-chip flash memory a read will return H'00, and writes are invalid.

The flash memory block configuration is shown in table 23.5.

EB4 (28 kbytes)	EB4 (28 kbytes)	H'(00)1000 to H'(00)7FFF
EB5 (16 kbytes)	EB5 (16 kbytes)	H'(00)8000 to H'(00)BFFF
EB6 (8 kbytes)	EB6 (8 kbytes)	H'(00)C000 to H'(00)DFFF
EB7 (8 kbytes)	EB7 (8 kbytes)	H'00E000 to H'00FFFF
EB8 (32 kbytes)	—	H'010000 to H'017FFF
EB9 (32 kbytes)	—	H'018000 to H'01FFFF

23.5.4 Serial/Timer Control Register (STCR)

Bit	7	6	5	4	3	2	1
	IICS	IICX1	IICX0	IICE	FLSHE	—	ICKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

STCR is an 8-bit readable/writable register that controls register access, the IIC operati (when the on-chip IIC option is included), and on-chip flash memory, and also selects t input clock. For details on functions not related to on-chip flash memory, see section 3. Serial/Timer Control Register (STCR), and descriptions of individual modules. If a mo controlled by STCR is not used, do not write 1 to the corresponding bit.

STCR is initialized to H'00 by a reset and in hardware standby mode.

Bits 7 to 4—I^{2}C Control (IICS, IICX1, IICX0, IICE): These bits control the operatio bus interface for the I^{2}C on-chip option. For details, see section 16, I^{2}C Bus Interface.

Bit 2—Reserved: Do not write 1 to this bit.

Bits 1 and 0—Internal Clock Select 1 and 0 (ICKS1, ICKS0): These bits control 8-
operation. See section 12, 8-Bit Timers, for details.

23.6 On-Board Programming Modes

When pins are set to on-board programming mode, program/erase/verify operations ca
performed on the on-chip flash memory. There are two on-board programming modes
and user program mode. The pin settings for transition to each of these modes are sho
23.6. For a diagram of the transitions to the various flash memory modes, see figure 2

Only advanced mode setting is possible for boot mode.

In the case of user program mode, established in advanced mode or normal mode, dep
the setting of the MD0 pin. In normal mode, only programming of a 56-kbyte area of
memory is possible.

Table 23.6 Setting On-Board Programming Modes

	Mode				
Mode Name	**CPU Operating Mode**	**MD1**	**MD0**	**P92**	**P91**
Boot mode	Advanced mode	0	0	1*	1*
User program mode	Advanced mode	1	0	—	—
	Normal mode		1	—	—

Note: * Can be used as I/O ports after boot mode is initiated.

completed, control branches to the start address of the programming control program a
programming control program execution state is entered (flash memory programming i
performed).

The transferred programming control program must therefore include coding that follo
programming algorithm given later.

The system configuration in boot mode is shown in figure 23.7, and the boot program r
execution procedure in figure 23.8.

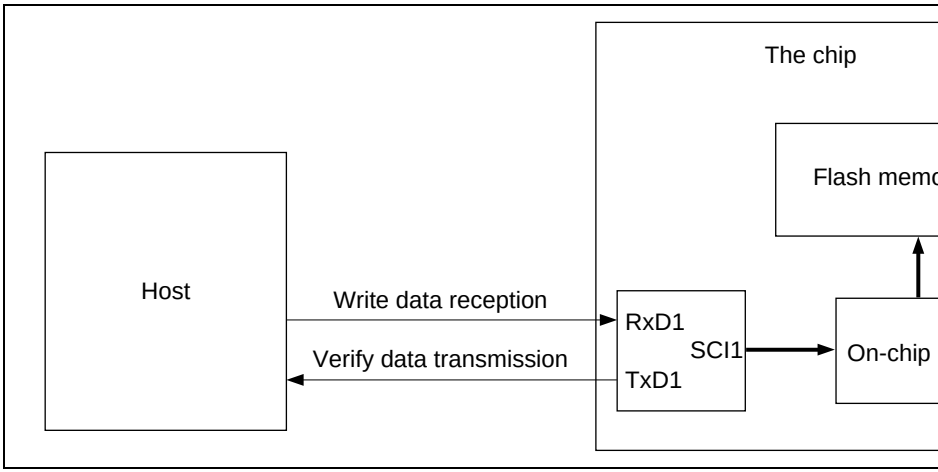
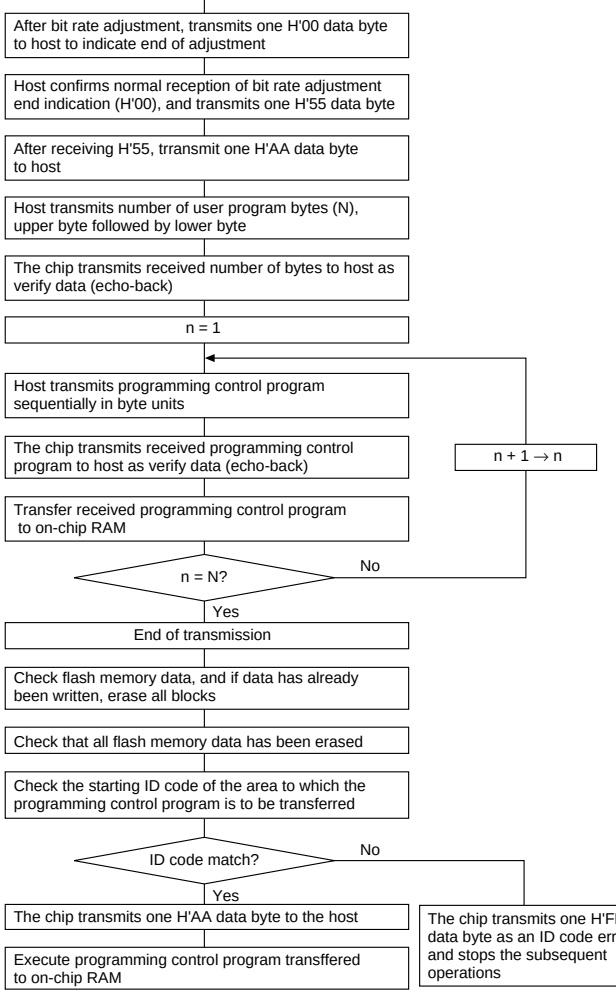


Figure 23.7 System Configuration in Boot Mode



Note: If a memory cell does not operate normally and cannot be erased, one H'FF byte is transmitted as an erase error
the erase operation and subsequent operations are halted.

Figure 23.8 Boot Mode Execution Procedure

Figure 23.9 Automatic SCI Bit Rate Adjustment

When boot mode is initiated, the chip measures the low period of the asynchronous SC communication data (H'00) transmitted continuously from the host. The SCI transmit/r format should be set as follows: 8-bit data, 1 stop bit, no parity. The chip calculates the the transmission from the host from the measured low period, and transmits one H'00 b host to indicate the end of bit rate adjustment. The host should confirm that this adjustn indication (H'00) has been received normally, and transmit one H'55 byte to the chip. If cannot be performed normally, initiate boot mode again (reset), and repeat the above op Depending on the host's transmission bit rate and the chips system clock frequency, the discrepancy between the bit rates of the host and the chip. To ensure correct SCI operat host's transfer bit rate should be set to (4800, 9600, or 19200) bps.

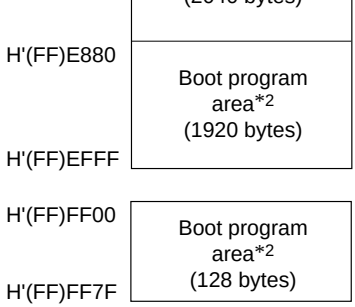
Table 23.7 shows typical host transfer bit rates and system clock frequencies for which adjustment of the chips bit rate is possible. The boot program should be executed withi system clock range.

Table 23.7 System Clock Frequencies for Which Automatic Adjustment of the C Rate Is Possible

Host Bit Rate	System Clock Frequency for Which Automatic Adjustm of Bit Rate Is Possible
19200 bps	8 MHz to 20 MHz
9600 bps	4 MHz to 20 MHz
4800 bps	2 MHz to 18 MHz

On-Chip RAM Area Divisions in Boot Mode

In boot mode, the 1920-byte area from H'(FF)E880 to H'(FF) EFFF and the 128-byte ar H'(FF)FF00 to H'(FF)FF7F is reserved for use by the boot program, as shown in figure area to which the programming control program is transferred is H'(FF)E080 to H'(FF) bytes). In the 64-kbyte version, this is a reserved area that is used only during the boot However, the 8-byte area from H'(FF)E080 to H'(FF)E087 is reserved for ID codes as s



Notes: 1. In the 64-kbyte version, this is a reserved area that is used only during the boot m
Do not use this area for other purposes.
2. The boot program area cannot be used until a transition is made to the execution
for the programming control program transferred to the RAM. Note that the boot p
remains stored in this area after a branch is made to the programming control pro

Figure 23.10 RAM Areas in Boot Mode

In the boot mode of this chip, the content in the 8-byte ID code area shown below is c
that whether or not there is a programming control program that corresponds to the ch
checked.

H'(FF)E080	40	FE	64	66	32	31	34
	↑ (product identification ID)						
H'(FF)E088 and above	Instruction code for write control program						

When a new programming control program for use in boot mode is created, add the 8-
code described above to the head of the program.

Notes on Use of Boot Mode

- When the chip comes out of reset in boot mode, it measures the low period of the i
SCI's RxD1 pin. The reset should end with RxD1 high. After the reset ends, it take
states for the chip to get ready to measure the low period of the RxD1 input.

- Before branching to the programming control program (RAM area H'(FF)E088), th terminates transmit and receive operations by the on-chip SCI (channel 1) (by clear and TE bits in SCR to 0), but the adjusted bit rate value remains set in BRR. The tra output pin, TxD1, goes to the high-level output state (P84DDR = 1, P84DR = 1).

  The contents of the CPU's internal general registers are undefined at this time, so th registers must be initialized immediately after branching to the programming contro In particular, since the stack pointer (SP) is used implicitly in subroutine calls, etc., must be specified for use by the programming control program.

  The initial values of other on-chip registers are not changed.

- Boot mode can be entered by making the pin settings shown in table 23.6 and execu reset-start.

  When the chip detects the boot mode setting at reset release*1, P92, P91, and P90 ca as I/O ports.

  Boot mode can be cleared by driving the reset pin low, waiting at least 20 states, the the mode pins, and executing reset release*1. Boot mode can also be cleared by a W overflow reset. The mode pin input levels must not be changed in boot mode.

- If the mode pin input levels are changed (for example, from low to high) during a re state of ports with multiplexed address functions and bus control output pins ($\overline{AS}$, $\overline{R}$ will change according to the change in the microcomputer's operating mode*2.

  Therefore, care must be taken to make pin settings to prevent these pins from becon signal pins during a reset, or to prevent collision with signals outside the microcomp

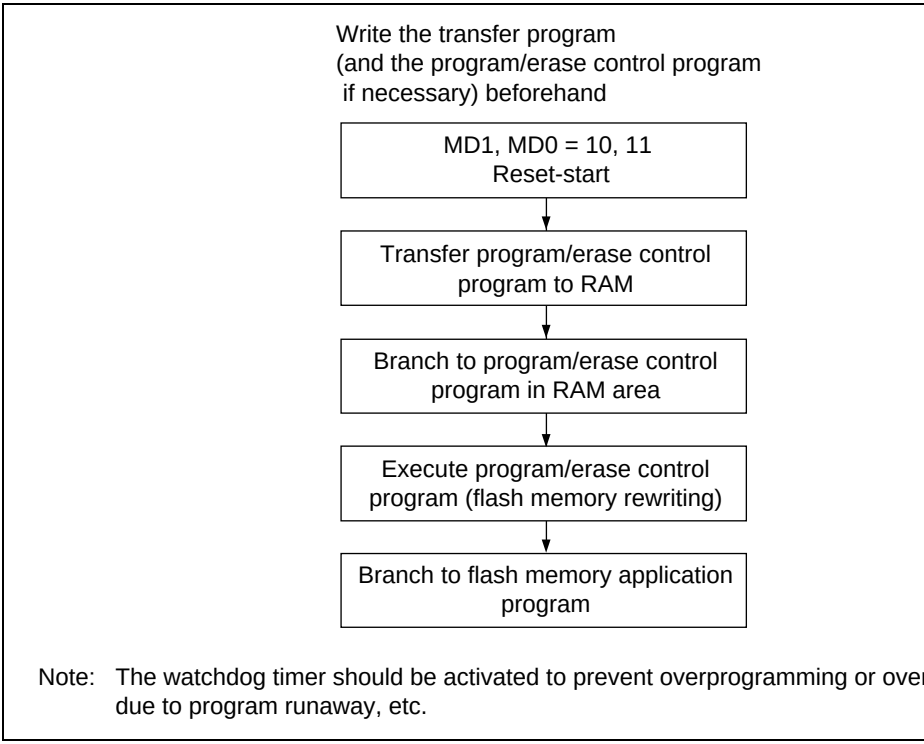
Notes: 1. Mode pins input must satisfy the mode programming setup time (t_{MDS} = 4 sta respect to the reset release timing.

2. Ports with multiplexed address functions will output a low level as the addre mode pin setting is for mode 1 is entered during a reset. In other modes, the go to the high-impedance state. The bus control output signals will output a if mode pin setting is for mode 1 is entered during a reset. In other modes, th go to the high-impedance state.

would in mode 2 and 3.

The flash memory itself cannot be read while the SWE bit is set to 1 to perform progr
erasing, so the control program that performs programming and erasing should be run
RAM or external memory.

Figure 23.11 shows the procedure for executing the program/erase control program wh
transferred to on-chip RAM.



Note: The watchdog timer should be activated to prevent overprogramming or over
due to program runaway, etc.

Figure 23.11 User Program Mode Execution Procedure

located and executed in on-chip RAM or external memory.

Notes: 1. Operation is not guaranteed if setting/resetting of the SWE, EV, PV, E, and
FLMCR1, and the ESU and PSU bits in FLMCR2, is executed by a program
memory.
2. Perform programming in the erased state. Do not perform additional progra
previously programmed addresses.

23.7.1 Program Mode

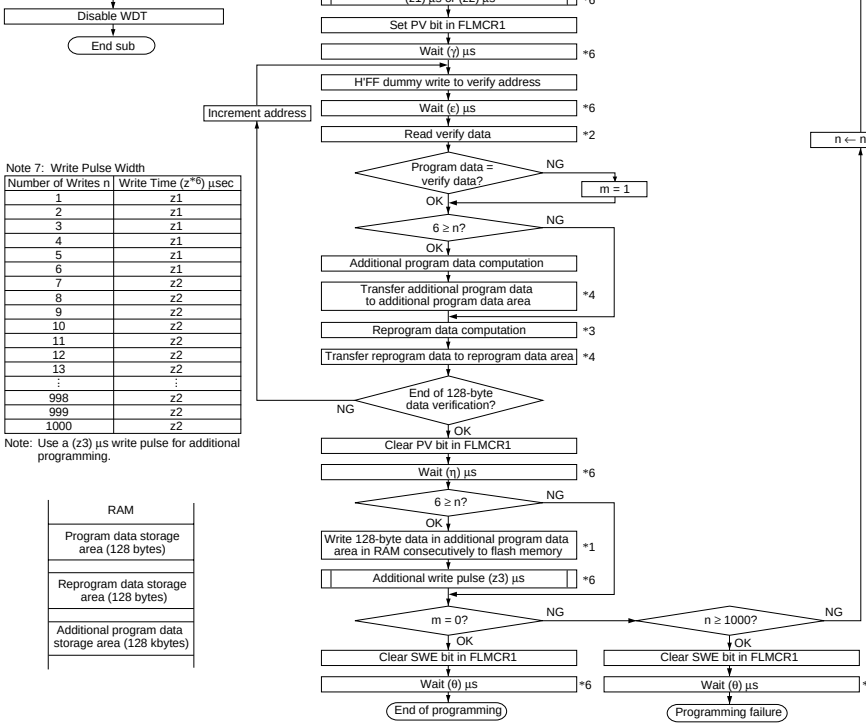
Follow the procedure shown in the program/program-verify flowchart in figure 23.12 t
or programs to flash memory. Performing program operations according to this flowcha
enable data or programs to be written to flash memory without subjecting the device to
stress or sacrificing program data reliability. Programming should be carried out 128 by
time.

The wait times (x, y, z1, z2, z3, α, ß, γ, ε, η, θ) after setting/clearing individual bits in f
memory control registers 1 and 2 (FLMCR1, FLMCR2) and the maximum number of v
see section 26.2.6, Flash Memory Characteristics.

Following the elapse of (x) µs or more after the SWE bit is set to 1 in flash memory co
register 1 (FLMCR1), 128-byte program data is stored in the program data area and rep
data area, and the 128-byte data in the reprogram data area written consecutively to the
addresses. The lower 8 bits of the first address written to must be H'00 or H'80. 128 co
byte data transfers are performed. The program address and program data are latched in
memory. A 128-byte data transfer must be performed even if writing fewer than 128 by
case, H'FF data must be written to the extra addresses.

Next, the watchdog timer is set to prevent overprogramming in the event of program ru
Set a value greater than (y + z2 + α + β) µs as the WDT overflow period. After this, pr
for program mode (program setup) is carried out by setting the PSU bit in FLMCR2, ar
elapse of (y) µs or more, the operating mode is switched to program mode by setting th
FLMCR1. The time during which the P bit is set is the flash memory programming tim

After the elapse of a given programming time, the programming mode is exited (the P
FLMCR1 is cleared, then the PSU bit in FLMCR2 is cleared at least (α) µs later). The
timer is cleared after the elapse of (β) µs or more, and the operating mode is switched
verify mode by setting the PV bit in FLMCR1. Before reading in program-verify mod
write of H'FF data should be made to the addresses to be read. The dummy write shou
executed after the elapse of (γ) µs or more. When the flash memory is read in this stat
is read in 16-bit units), the data at the latched address is read. Wait at least (ε) µs after
write before performing this read operation. Next, the originally written data is compa
verify data, and reprogram data is computed (see figure 23.12) and transferred to the r
data area. After 128 bytes of data have been verified, exit program-verify mode, wait f
(η) µs. If the programming count is less than 6, the 128-byte data in the additional pro
area should be written consecutively to the write addresses, and additional programmi
performed. Next clear the SWE bit in FLMCR1, and wait at least (θ) µs. If reprogram
necessary, set program mode again, and repeat the program/program-verify sequence
However, ensure that the program/program-verify sequence is not repeated more than
the same bits.



Note 7: Write Pulse Width

Number of Writes n	Write Time (z*6) µsec
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 performe if writing fewer than 128 bytes; in this case, H'FF 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 in the 128-byte programming loop will be subjected to additional programming if they fail the subsec verify operation.
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 program data must be pro RAM. The reprogram and additional program data contents are modified as programming proceeds.
5. The write pulse of (z1) µs or (z2) µs is applied according to the progress of the programming operation. See Note 7 for the pulse widths. When writing of a program data is executed, a (z3) µs write pulse should be applied. Reprogram data X means reprogram data when the write pulse is applied.
6. See section 26.2.6, Flash Memory Characteristics, for the values of x, y, z1, z2, z3, α, β, γ, ε, η, θ, and N.

Program Data Computation Chart

Original Data (D)	Verify Data (V)	Reprogram Data (X)	Comments
0	0	1	Programming completed
	1	0	Programming incomplete; reprogram
1	0	1	
	1	1	Still in erased state; no action

Additional Program Data Computation Chart

Reprogram Data (X')	Verify Data (V)	Additional Program Data (Y)	Comments
0	0	0	Additional programming executed
	1	1	Additional programming not executed
1	0	1	
	1	1	Additional programming not executed

Figure 23.12 Program/Program-Verify Flowchart

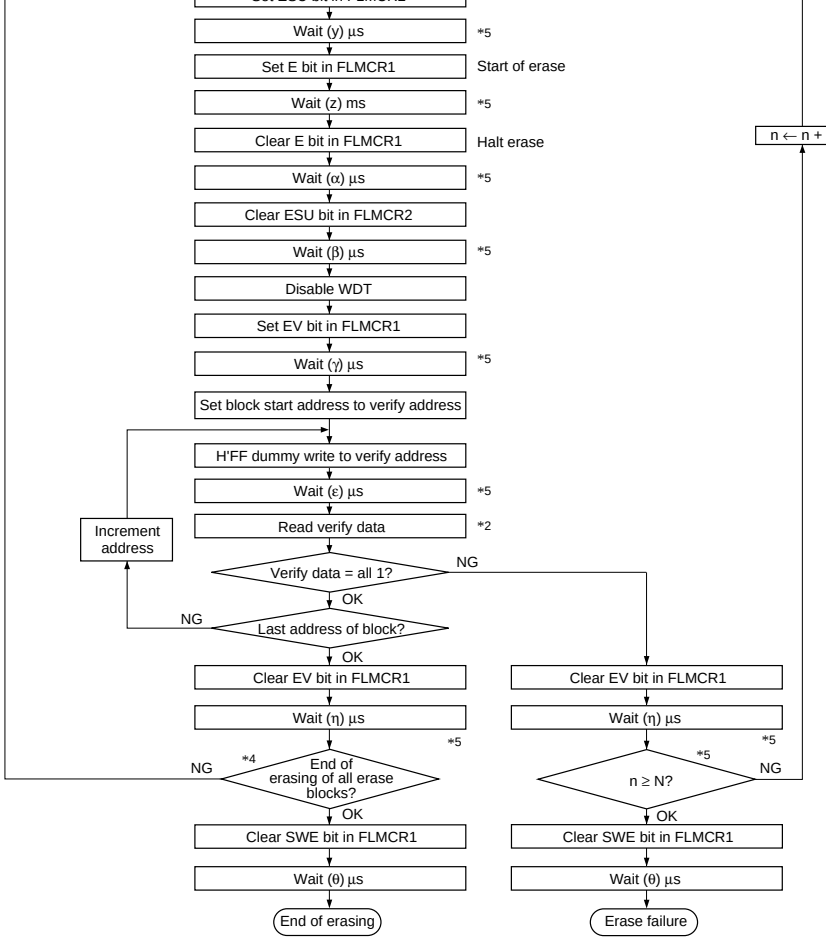
To perform data or program erasure, make a 1 bit setting for the flash memory area to erase block register 1 or 2 (EBR1 or EBR2) at least (x) µs after setting the SWE bit to memory control register 1 (FLMCR1). Next, the watchdog timer is set to prevent over the event of program runaway, etc. Set a value greater than (y + z + α + β) ms as the W overflow period. After this, preparation for erase mode (erase setup) is carried out by s ESU bit in FLMCR2, and after the elapse of (y) µs or more, the operating mode is swi erase mode by setting the E bit in FLMCR1. The time during which the E bit is set is t memory erase time. Ensure that the erase time does not exceed (z) ms.

Note: With flash memory erasing, preprogramming (setting all data in the memory t to 0) is not necessary before starting the erase procedure.

23.7.4 Erase-Verify Mode

In erase-verify mode, data is read after memory has been erased to check whether it ha correctly erased.

After the elapse of the erase time, erase mode is exited (the E bit in FLMCR1 is cleare ESU bit in FLMCR2 is cleared at least (α) µs later), the watchdog timer is cleared afte of (β) µs or more, and the operating mode is switched to erase-verify mode by setting FLMCR1. Before reading in erase-verify mode, a dummy write of H'FF data should b the addresses to be read. The dummy write should be executed after the elapse of (γ) µ When the flash memory is read in this state (verify data is read in 16-bit units), the dat latched address is read. Wait at least (ε) µs after the dummy write before performing t operation. If the read data has been erased (all 1), a dummy write is performed to the n and erase-verify is performed. If the read data has not been erased, set erase mode aga repeat the erase/erase-verify sequence in the same way. However, ensure that the erase verify sequence is not repeated more than (N) times. When verification is completed, verify mode, and wait for at least (η) µs. If erasure has been completed on all the erase clear the SWE bit in FLMCR1, and wait (θ) µs. If there are any unerased blocks, make setting in EBR1 or EBR2 for the flash memory area to be erased, and repeat the erase/ sequence in the same way.



Notes: 1. Preprogramming (setting erase block data to all 0) is not necessary.
2. Verify data is read in 16-bit (W) units.
3. Set only one bit in EBR1or EBR2. More than one bit cannot be set.
4. Erasing is performed in block units. To erase a number of blocks, the individual blocks must be erased sequentially.
5. See section 26.2.6, Flash Memory Characteristics, for the values of x, y, z, α, β, γ, ε, η, θ, and N.

Figure 23.13 Erase/Erase-Verify Flowchart (Single-Block Erase)

disabled or aborted. Hardware protection is reset by settings in flash memory control r
and 2 (FLMCR1, FLMCR2) and erase block registers 1 and 2 (EBR1, EBR2). (See tab

Table 23.8 Hardware Protection

Item	Description	Func Program
Reset/standby protection	• In a reset (including a WDT overflow reset) and in hardware standby mode, software standby mode, subactive mode, subsleep mode, and watch mode, FLMCR1, FLMCR2, EBR1, and EBR2 are initialized, and the program/erase-protected state is entered. • 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, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width specified in the AC Characteristics section.	Yes

23.8.2 Software Protection

Software protection can be implemented by setting the SWE bit in FLMCR1 and erase
registers 1 and 2 (EBR1, EBR2). When software protection is in effect, setting the P o
flash memory control register 1 (FLMCR1) does not cause a transition to program mo
mode. (See table 23.9.)

protection	by settings in erase block registers 1 and 2 (EBR1, EBR2). • Setting EBR1 and EBR2 to H'00 places all blocks in the erase-protected state.

23.8.3 Error Protection

In error protection, an error is detected when MCU runaway occurs during flash memo programming/erasing, or operation is not performed in accordance with the program/er algorithm, and the program/erase operation is aborted. Aborting the program/erase oper prevents damage to the flash memory due to overprogramming or overerasing.

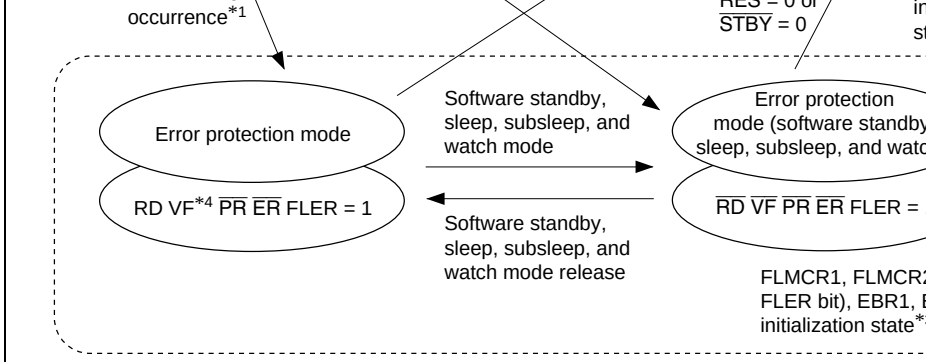
If the MCU malfunctions during flash memory programming/erasing, the FLER bit is s FLMCR2 and the error protection state is entered. The FLMCR1, FLMCR2, EBR1, an settings are retained, but program mode or erase mode is aborted at the point at which t occurred. Program mode or erase mode cannot be re-entered by re-setting the P or E bi PV and EV bit setting is enabled, and a transition can be made to verify mode.

FLER bit setting conditions are as follows:

- When flash memory is read during programming/erasing (including a vector read o fetch)
- Immediately after exception handling (excluding a reset) during programming/erasi
- When a SLEEP instruction (including software standby, sleep, subactive, subsleep a mode) is executed during programming/erasing
- When the bus is released during programming/erasing

Error protection is released only by a reset and in hardware standby mode.

Figure 23.14 shows the flash memory state transition diagram.



Legend:

RD: Memory read possible	$\overline{RD}$: Memory read not possible
VF: Verify-read possible	$\overline{VF}$: Verify-read not possible
PR: Programming possible	$\overline{PR}$: Programming not possible
ER: Erasing possible	$\overline{ER}$: Erasing not possible

Notes: 1. When an error occurs other than due to a SLEEP instruction, or when a SLEEP instruc executed for a transition to subactive mode
2. When an error occurs due to a SLEEP instruction (except subactive mode)
3. Except sleep mode
4. $\overline{VF}$ in subactive mode

Figure 23.14 Flash Memory State Transitions

2. In the interrupt exception handling sequence during programming or erasing, the ve not be read correctly*[2], possibly resulting in MCU runaway.
3. If interrupt occurred during boot program execution, it would not be possible to exe normal boot mode sequence.

For these reasons, in on-board programming mode alone there are conditions for disabl interrupt, as an exception to the general rule. However, this provision does not guarante erasing and programming or MCU operation. All interrupt requests, including NMI inp therefore be disabled inside and outside the MCU when programming or erasing flash m Interrupt is also disabled in the error-protection state while the P or E bit remains set in

Notes: 1. Interrupt requests must be disabled inside and outside the MCU until the pro control program has completed programming.
2. The vector may not be read correctly in this case for the following two reaso
   - If flash memory is read while being programmed or erased (while the P set in FLMCR1), correct read data will not be obtained (undetermined va be returned).
   - If the interrupt entry in the vector table has not been programmed yet, in exception handling will not be executed correctly.

and section 23.11, Flash Memory Programming and Erasing Precautions, for notes on
mode. Flash memory read mode, auto-program mode, auto-erase mode, and status rea
supported with these device types. In auto-program mode, auto-erase mode, and status
a status polling procedure is used, and in status read mode, detailed internal signals ar
execution of an auto-program or auto-erase operation.

Table 23.10 shows programmer mode pin settings.

Note: * Use products of the H8S/2148 A-mask version, H8S/2147 A-mask version
H8S/2144 A-mask version (in either 5-V or 3-V version) with the writing v
PROM programmer set to 3.3 V. Do not use products other than the A-mas
with 3.3V PROM programmer setting.

Table 23.10 Programmer Mode Pin Settings

Pin Names	Setting/External Circuit Connection
Mode pins: MD1, MD0	Low-level input to MD1, MD0
$\overline{\text{STBY}}$ pin	High-level input (Hardware standby mode n
$\overline{\text{RES}}$ pin	Power-on reset circuit
XTAL and EXTAL pins	Oscillation circuit
Other setting pins: P97, P92, P91, P90, P67	Low-level input to p92, p67, high-level input P91, P90

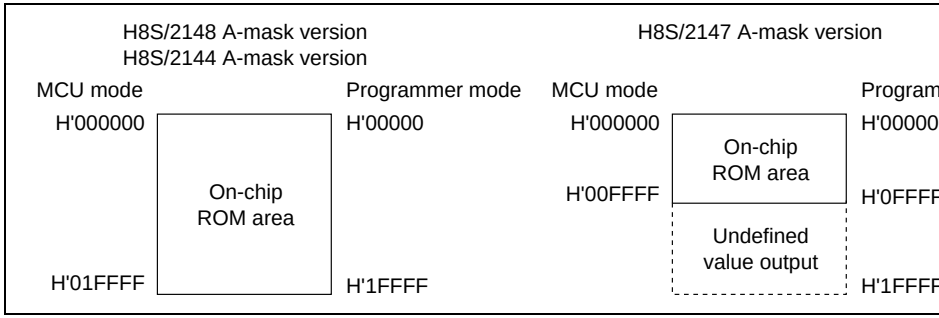


Figure 23.15 Memory Map in Programmer Mode

23.10.3 Programmer Mode Operation

Table 23.11 shows how the different operating modes are set when using programmer m
table 23.12 lists the commands used in programmer mode. Details of each mode are giv

- Memory Read Mode

  Memory read mode supports byte reads.
- Auto-Program Mode

  Auto-program mode supports programming of 128 bytes at a time. Status polling is
  confirm the end of auto-programming.
- Auto-Erase Mode

  Auto-erase mode supports automatic erasing of the entire flash memory. Status poll
  to confirm the end of auto-erasing.
- Status Read Mode

  Status polling is used for auto-programming and auto-erasing, and normal terminati
  confirmed by reading the FO6 signal. In status read mode, error information is outp
  error occurs.

Notes: 1. Chip disable is not a standby state; internally, it is an operation state.
2. Ain indicates that there is also address input in auto-program mode.

Table 23.12 Programmer Mode Commands

Command Name	Number of Cycles	1st Cycle			2nd Cy	
		Mode	Address	Data	Mode	Addres
Memory read mode	1 + n	Write	X	H'00	Read	RA
Auto-program mode	129	Write	X	H'40	Write	WA
Auto-erase mode	2	Write	X	H'20	Write	X
Status read mode	2	Write	X	H'71	Write	X

Notes: 1. In auto-program mode. 129 cycles are required for command writing by a s
128-byte write.
2. In memory read mode, the number of cycles depends on the number of ad
cycles (n).

23.10.4 Memory Read Mode

- After the end of an auto-program, auto-erase, or status read operation, the comman
is entered. To read memory contents, a transition must be made to memory read m
means of a command write before the read is executed.
- Command writes can be performed in memory read mode, just as in the command
- Once memory read mode has been entered, consecutive reads can be performed.
- After power-on, memory read mode is entered.

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

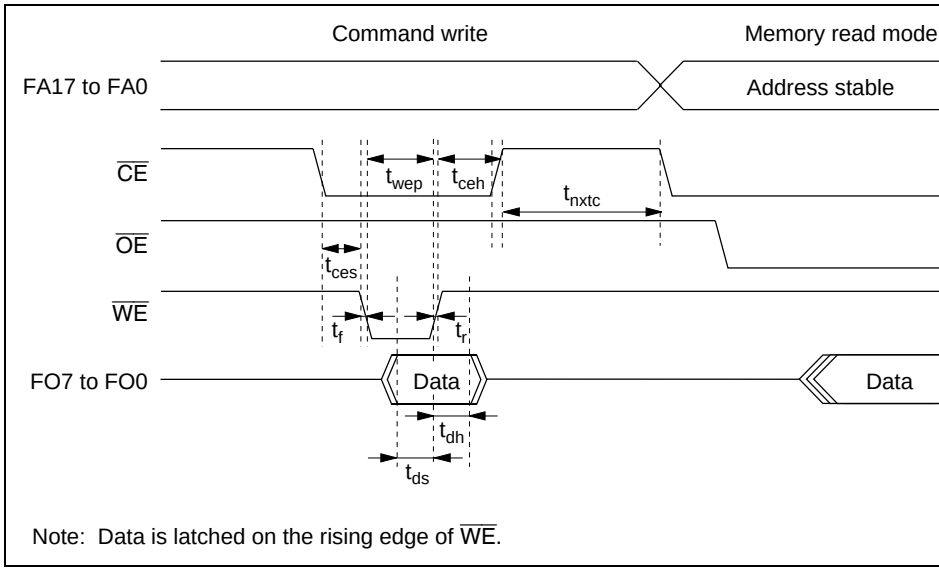


Figure 23.16 Memory Read Mode Timing Waveforms after Command W

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

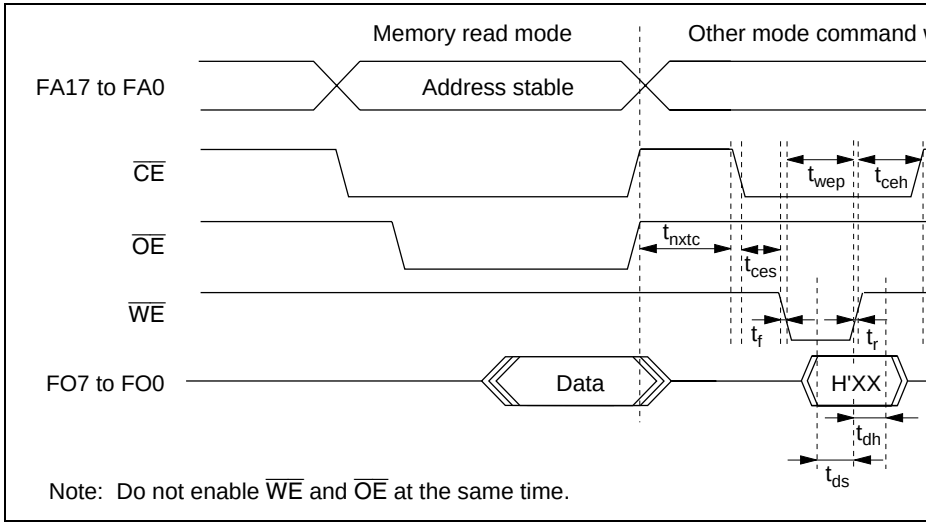


Figure 23.17 Timing Waveforms when Entering Another Mode from Memory

Output disable delay time	t_{df}	—	100	ns
Data output hold time	t_{oh}	5	—	ns

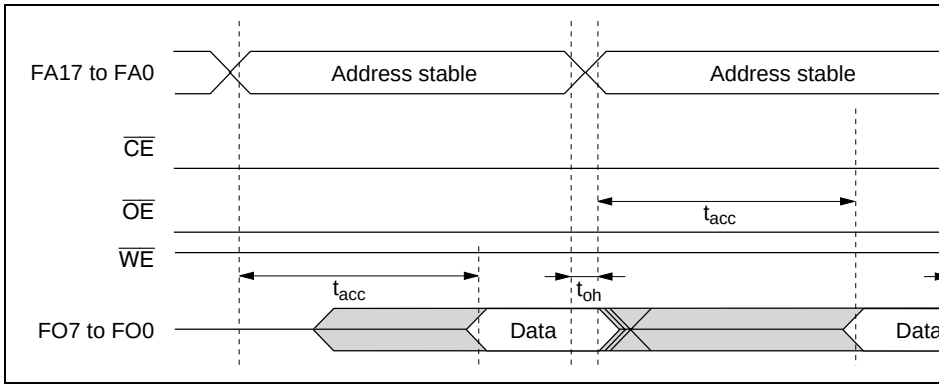


Figure 23.18 Timing Waveforms for $\overline{CE}$/$\overline{OE}$ Enable State Read

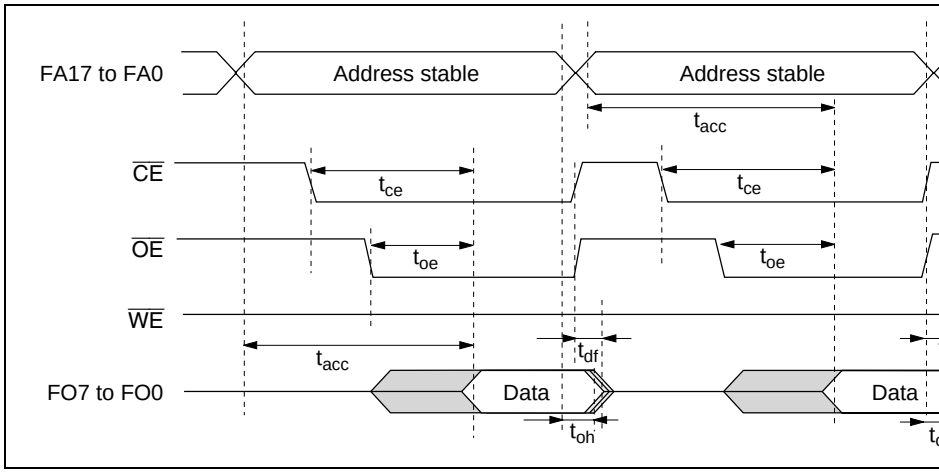


Figure 23.19 Timing Waveforms for $\overline{CE}$/$\overline{OE}$ Clocked Read

Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
Status polling start time	t_{wsts}	1	—	ms
Status polling access time	t_{spa}	—	150	ns
Address setup time	t_{as}	0	—	ns
Address hold time	t_{ah}	60	—	ns
Memory write time	t_{write}	1	3000	ms
$\overline{WE}$ rise time	t_{r}	—	30	ns
$\overline{WE}$ fall time	t_{f}	—	30	ns

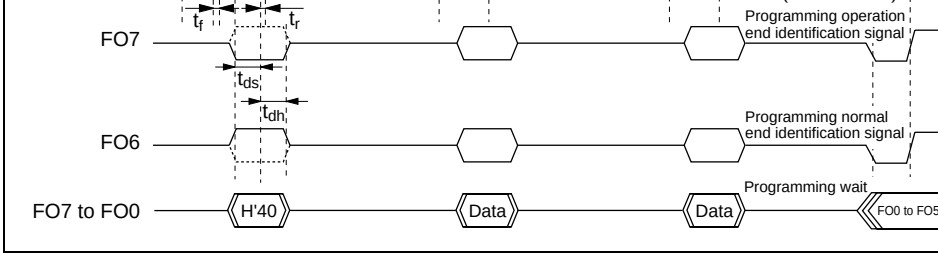


Figure 23.20 Auto-Program Mode Timing Waveforms

Notes on Use of Auto-Program Mode

- In auto-program mode, 128 bytes are programmed simultaneously. This should be c by executing 128 consecutive byte transfers.
- A 128-byte data transfer is necessary even when programming fewer than 128 bytes case, H'FF data must be written to the extra addresses.
- The lower 8 bits of the transfer address must be H'00 or H'80. If a value other than a address is input, processing will switch to a memory write operation but a write erro flagged.
- Memory address transfer is performed in the second cycle (figure 23.20). Do not pe transfer after the second cycle.
- Do not perform a command write during a programming operation.
- Perform one auto-programming operation for a 128-byte block for each address. Characteristics are not guaranteed for two or more programming operations.
- Confirm normal end of auto-programming by checking FO6. Alternatively, status re can also be used for this purpose (FO7 status polling uses the auto-program operatio identification pin).
- The status polling FO6 and FO7 pin information is retained until the next command Until the next command write is performed, reading is possible by enabling $\overline{CE}$ and

Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
Status polling start time	t_{ests}	1	—	ms
Status polling access time	t_{spa}	—	150	ns
Memory erase time	t_{erase}	100	40000	ms
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

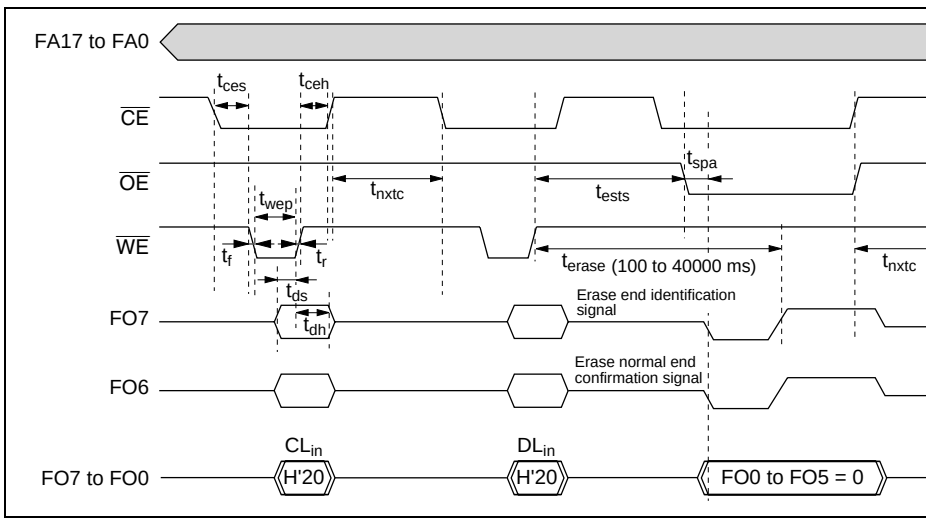


Figure 23.21 Auto-Erase Mode Timing Waveforms

Until the next command write is performed, reading is possible by enabling CE and

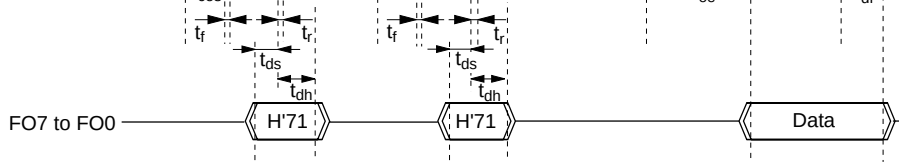
23.10.7 Status Read Mode

- Status read mode is used to identify what type of abnormal end has occurred. Use th when an abnormal end occurs in auto-program mode or auto-erase mode.
- The return code is retained until a command write for other than status read mode is performed.

Table 23.18 AC Characteristics in Status Read Mode

Conditions: V_{CC} = 3.3 V ±0.3 V, V_{SS} = 0 V, T_a = 25°C ±5°C

Item	Symbol	Min	Max	Unit
Command write cycle	t_{nxtc}	20	—	µs
$\overline{CE}$ hold time	t_{ceh}	0	—	ns
$\overline{CE}$ setup time	t_{ces}	0	—	ns
Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{OE}$ output delay time	t_{oe}	—	150	ns
Disable delay time	t_{df}	—	100	ns
$\overline{CE}$ output delay time	t_{ce}	—	150	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns



Note: FO2 and FO3 are undefined.

Figure 23.22 Status Read Mode Timing Waveforms

Table 23.19 Status Read Mode Return Commands

Pin Name	FO7	FO6	FO5	FO4	FO3	FO2	FO1
Attribute	Normal end identification	Command error	Program-ming error	Erase error	—	—	Program-ming or erase count exceeded
Initial value	0	0	0	0	0	0	0
Indications	Normal end: 0 Abnormal end: 1	Command error: 1 Otherwise: 0	Program-ming error: 1 Otherwise: 0	Erase error: 1 Otherwise: 0	—	—	Count exceeded: 1 Otherwise: 0

Note: FO2 and FO3 are undefined.

23.10.8 Status Polling

- The FO7 status polling flag indicates the operating status in auto-program or auto-
- The FO6 status polling flag indicates a normal or abnormal end in auto-program o mode.

23.10.9 Programmer Mode Transition Time

Commands cannot be accepted during the oscillation stabilization period or the program
setup period. After the programmer mode setup time, a transition is made to memory re

Table 23.21 Command Wait State Transition Time Specifications

Item	Symbol	Min	Max	Unit
Standby release (oscillation stabilization time)	t_{osc1}	20	—	ms
Programmer mode setup time	t_{bmv}	10	—	ms
V_{CC} hold time	t_{dwn}	0	—	ms

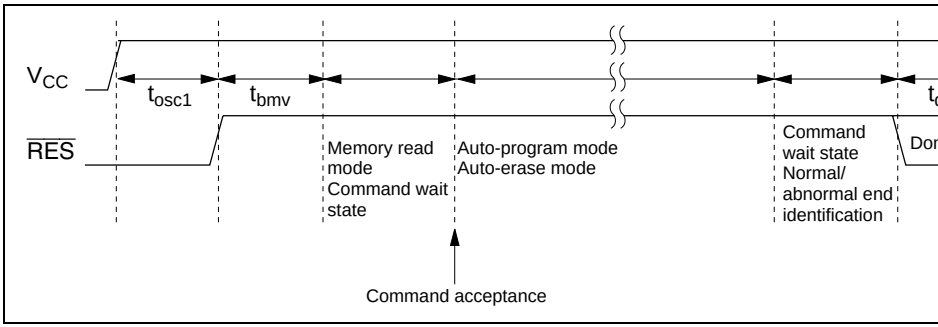


**Figure 23.23 Oscillation Stabilization Time, Programmer Mode Setup Time, ar
Supply Fall Sequence**

Notes: 1. The flash memory is initially in the erased state when the device is shippe
For other chips for which the erasure history is unknown, it is recommende
erasing be executed to check and supplement the initialization (erase) level
2. Auto-programming should be performed once only on the same address bl

23.11 Flash Memory Programming and Erasing Precautions

Precautions concerning the use of on-board programming mode and programmer mod
summarized below.

Use the specified voltages and timing for programming and erasing.
Applied voltages in excess of the rating can permanently damage the device. For a PR
programmer, use Renesas Technology microcomputer device types with 128-kbyte or
on-chip flash memory that support a 3.3-V programming voltage.
Do not select the HN28F101, or use a programming voltage of 5.0 V for the PROM p
and only use the specified socket adapter. Incorrect use will result in damaging the de

Powering on and off
When applying or disconnecting V_{CC}, fix the $\overline{RES}$ pin low and place the flash memory
hardware protection state.
The power-on and power-off timing requirements should also be satisfied in the event
failure and subsequent recovery.

Use the recommended algorithm when programming and erasing flash memory.
The recommended algorithm enables programming and erasing to be carried out witho
subjecting the device to voltage stress or sacrificing program data reliability. When se
E bit in FLMCR1, the watchdog timer should be set beforehand as a precaution agains
runaway, etc.

All interrupt requests, including NMI, should be disabled when programming and erasi
memory to give priority to program/erase operations.

Do not perform additional programming. Erase the memory before reprogrammi
In on-board programming, perform only one programming operation on a 128-byte pro
unit block. In programmer mode, too, perform only one programming operation on a 12
programming unit block. Programming should be carried out with the entire programm
block erased.

Before programming, check that the chip is correctly mounted in the PROM prog
Overcurrent damage to the device can result if the index marks on the PROM programr
socket adapter, and chip are not correctly aligned.

Do not touch the socket adapter or chip during programming.
Touching either of these can cause contact faults and write errors.

23.12 Note on Switching from F-ZTAT Version to Mask ROM Ver

The mask ROM version does not have the internal registers for flash memory control th
provided in the F-ZTAT version. Table 23.22 lists the registers that are present in the F
version but not in the mask ROM version. If a register listed in table 23.22 is read in th
ROM version, an undefined value will be returned. Therefore, if application software d
on the F-ZTAT version is switched to a mask ROM version product, it must be modifie
that the registers in table 23.22 have no effect

Table 23.22 Registers Present in F-ZTAT Version but Absent in Mask ROM Vers

Register	Abbreviation	Address
Flash memory control register 1	FLMCR1	H'FF80
Flash memory control register 2	FLMCR2	H'FF81
Erase block register 1	EBR1	H'FF82
Erase block register 2	EBR2	H'FF83

The clock pulse generator consists of an oscillator circuit, a duty adjustment circuit, cl
selection circuit, medium-speed clock divider, bus master clock selection circuit, subc
circuit, and waveform shaping circuit.

24.1.1 Block Diagram

Figure 24.1 shows a block diagram of the clock pulse generator.

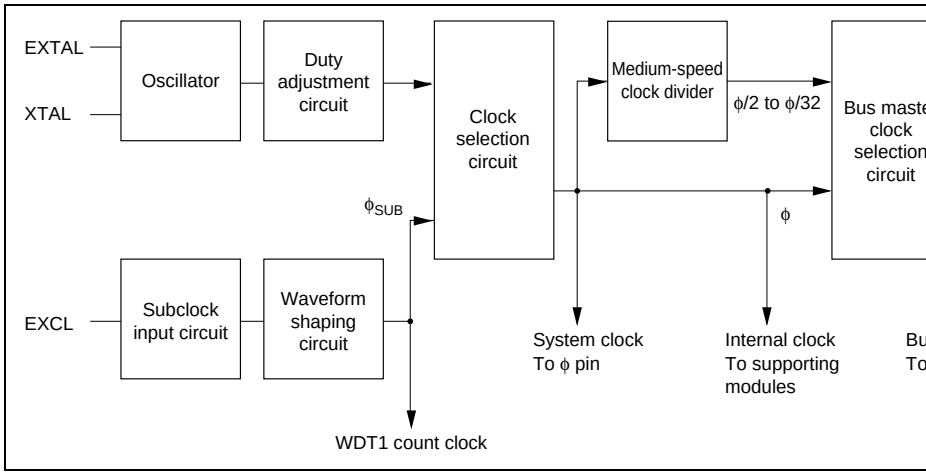


Figure 24.1 Block Diagram of Clock Pulse Generator

Low-power control register	LPWRCR	R/W	H'00	H'FF8

Note: * Lower 16 bits of the address.

24.2 Register Descriptions

24.2.1 Standby Control Register (SBYCR)

Bit	7	6	5	4	3	2	1	
	SSBY	STS2	STS1	STS0	—	SCK2	SCK1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	—	R/W	R/W	

SBYCR is an 8-bit readable/writable register that performs power-down mode control.

Only bits 0 to 2 are described here. For a description of the other bits, see section 25.2.1 Control Register (SBYCR).

SBYCR is initialized to H'00 by a reset and in hardware standby mode. It is not initializ software standby mode.

Bits 2 to 0—System Clock Select 2 to 0 (SCK2 to SCK0): These bits select the bus m for high-speed mode and medium-speed mode.

When operating the device after a transition to subactive mode or watch mode bits SCK should all be cleared to 0.

	1	Medium-speed clock is ϕ/32	
1	—	—	

24.2.2 Low-Power Control Register (LPWRCR)

Bit	7	6	5	4	3	2	1
	DTON	LSON	NESEL	EXCLE	—	—	—
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	—	—	—

LPWRCR is an 8-bit readable/writable register that performs power-down mode contr

Only bit 4 is described here. For a description of the other bits, see section 25.2.2, Low Control Register (LPWRCR).

LPWRCR is initialized to H'00 by a reset and in hardware standby mode. It is not initi software standby mode.

Bit 4—Subclock Input Enable (EXCLE): Controls subclock input from the EXCL p

Bit 4

EXCLE	Description	
0	Subclock input from EXCL pin is disabled	(I
1	Subclock input from EXCL pin is enabled	

A crystal resonator can be connected as shown in the example in figure 24.2. Select the resistance R_d according to table 24.2. An AT-cut parallel-resonance crystal should be us

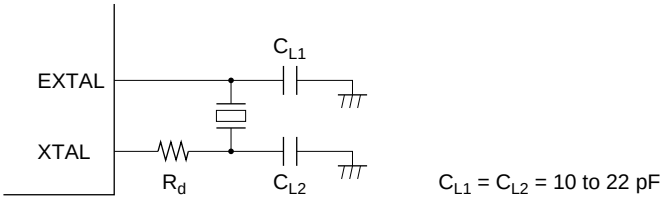


Figure 24.2 Connection of Crystal Resonator (Example)

Table 24.2 Damping Resistance Value

Frequency (MHz)	2	4	8	10	12	16
R_d (Ω)	1k	500	200	0	0	0

Crystal resonator

Figure 24.3 shows the equivalent circuit of the crystal resonator. Use a crystal resonato the characteristics shown in table 24.3 and the same frequency as the system clock (ϕ).

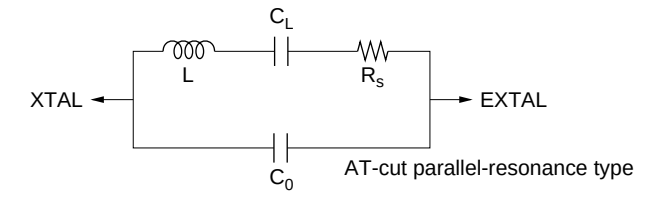


Figure 24.3 Crystal Resonator Equivalent Circuit

When a crystal resonator is connected, the following points should be noted.

Other signal lines should be routed away from the oscillator circuit to prevent inductio
interfering with correct oscillation. See figure 24.4.

When designing the board, place the crystal resonator and its load capacitors as close
to the XTAL and EXTAL pins.

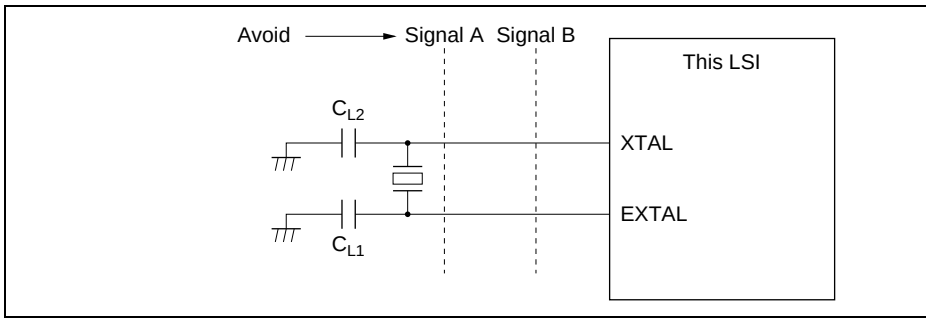


Figure 24.4 Example of Incorrect Board Design

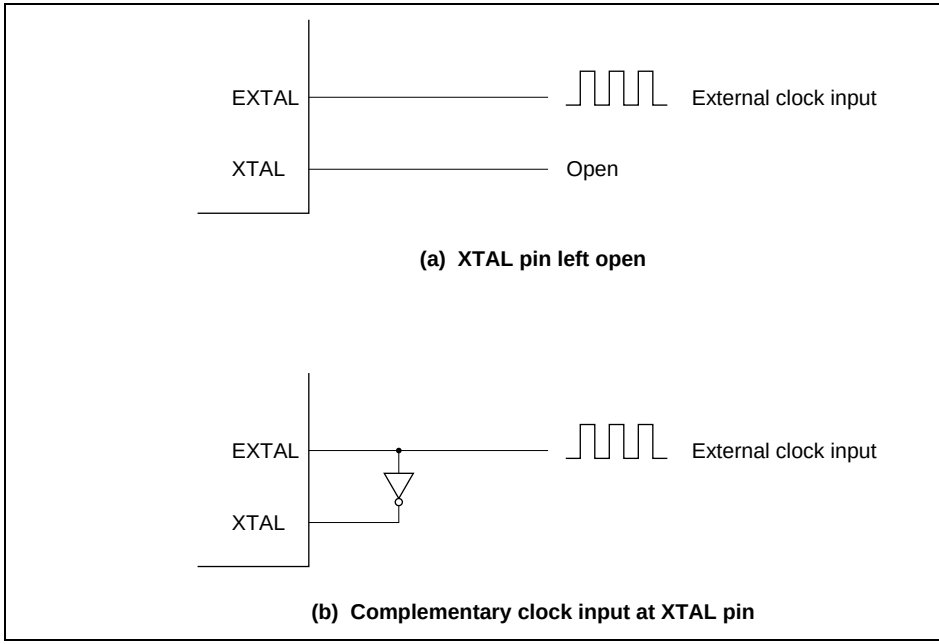


Figure 24.5 External Clock Input (Examples)

Item	Symbol	Min	Max	Min	Max	Unit	Test Cond
External clock input low pulse width	t_{EXL}	40	—	20	—	ns	Figure 24.6
External clock input high pulse width	t_{EXH}	40	—	20	—	ns	
External clock rise time	t_{EXr}	—	10	—	5	ns	
External clock fall time	t_{EXf}	—	10	—	5	ns	
Clock low pulse width	t_{CL}	0.4	0.6	0.4	0.6	t_{cyc}	$\phi \geq 5$ MHz
		80	—	80	—	ns	$\phi < 5$ MHz
Clock high pulse width	t_{CH}	0.4	0.6	0.4	0.6	t_{cyc}	$\phi \geq 5$ MHz
		80	—	80	—	ns	$\phi < 5$ MHz

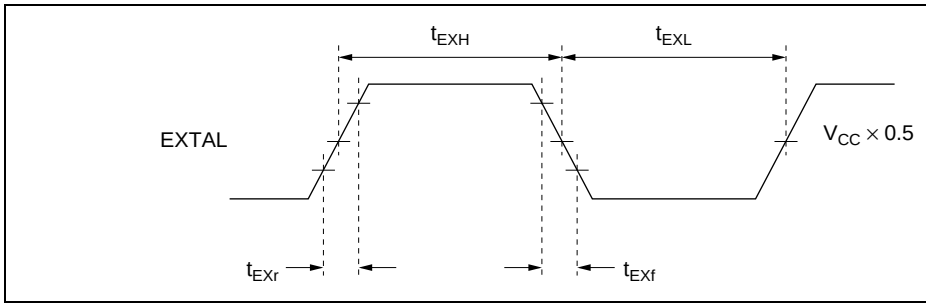


Figure 24.6 External Clock Input Timing

Table 24.5 shows the external clock output settling delay time, and figure 24.7 shows clock output settling delay timing. The oscillator and duty adjustment circuit have a fu adjusting the waveform of the external clock input at the EXTAL pin. When the presc signal is input at the EXTAL pin, internal clock signal output is fixed after the elapse

output settling
delay time

Note: * t_{DEXT} includes $\overline{RES}$ pulse width (t_{RESW}).

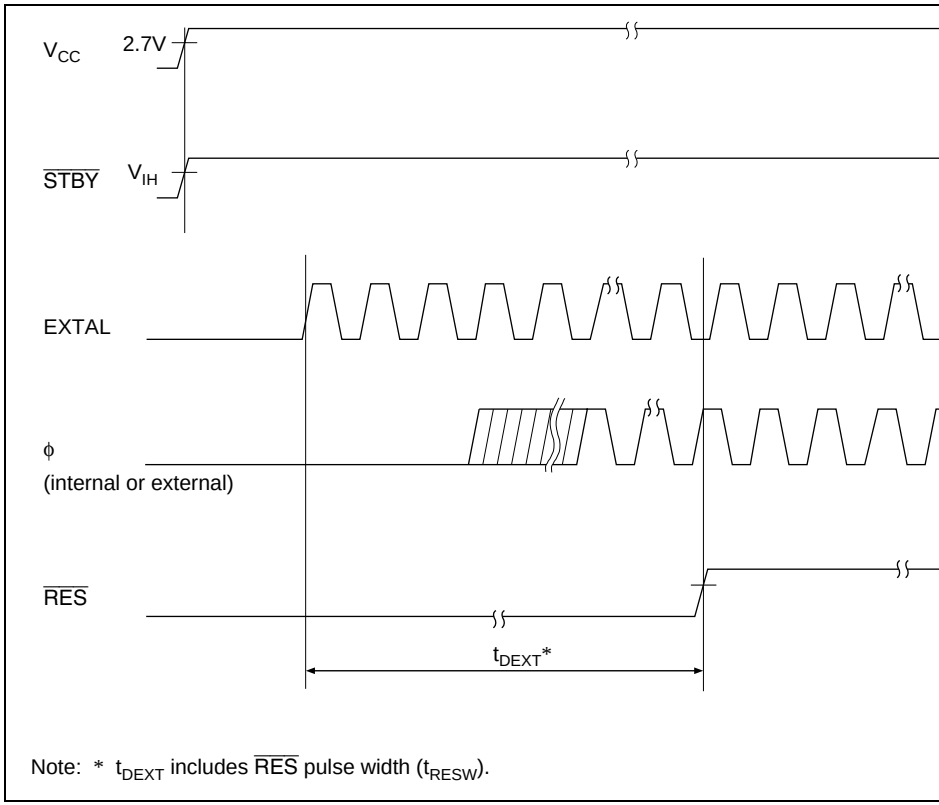


Figure 24.7 External Clock Output Settling Delay Timing

clocks.

24.6 Bus Master Clock Selection Circuit

The bus master clock selection circuit selects the system clock (ϕ) or one of the mediu
clocks (ϕ/2, ϕ/4, ϕ/8, ϕ/16, or ϕ/32) to be supplied to the bus master, according to the s
bits SCK2 to SCK0 in SBYCR.

24.7 Subclock Input Circuit

The subclock input circuit controls the subclock input from the EXCL pin.

Inputting the Subclock

When a subclock is used, a 32.768 kHz external clock should be input from the EXCL
case, clear bit P96DDR to 0 in P9DDR and set bit EXCLE to 1 in LPWRCR.

The subclock input conditions are shown in table 24.6 and figure 24.8.

Table 24.6 Subclock Input Conditions

		V_{cc} = 2.7 to 5.5 V				
Item	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test C**
Subclock input low pulse width	t_{EXCLL}	—	15.26	—	µs	Figure
Subclock input high pulse width	t_{EXCLH}	—	15.26	—	µs	
Subclock input rise time	t_{EXCLr}	—	—	10	ns	
Subclock input fall time	t_{EXCLf}	—	—	10	ns	

Figure 24.8 Subclock Input Timing

When Subclock Is Not Needed

Do not enable subclock input when the subclock is not needed

Note on Subclock Usage

In transiting to power-down mode, if at least two cycles of the 32-kHz clock are not inp
32-kHz clock input is enabled (EXCLE = 1) until the SLEEP instruction is executed (p
mode transition), the subclock input circuit is not initialized and an error may occur in t
microcomputer.

Before power-down mode is entered with using the subclock, at least two cycle of the 3
clock should be input after the 32-kHz clock input is enabled (EXCLE = 1).

As described in the hardware manual (clock pulse generator/subclock input circuit), wh
subclock is not used, the subclock input should not be enabled (EXCLE = 0).

24.8 Subclock Waveform Shaping Circuit

To eliminate noise in the subclock input from the EXCL pin, this circuit samples the cl
clock obtained by dividing the ϕ clock. The sampling frequency is set with the NESEL
LPWRCR. For details, see sections 24.2.2 and 25.2.2, Low-Power Control Register (LP
The clock is not sampled in subactive mode, subsleep mode, or watch mode.

is selected as the system clock. In these modes, modules such as CPU, TMR0, TMR1, WDT1, and I/O ports operate on the ϕ SUB clock. The count clock for each timer is a obtained by driving the ϕ SUB clock.

so on.

This LSI has the following operating modes:

1. High-speed mode
2. Medium-speed mode
3. Subactive mode
4. Sleep mode
5. Subsleep mode
6. Watch mode
7. Module stop mode
8. Software standby mode
9. Hardware standby mode

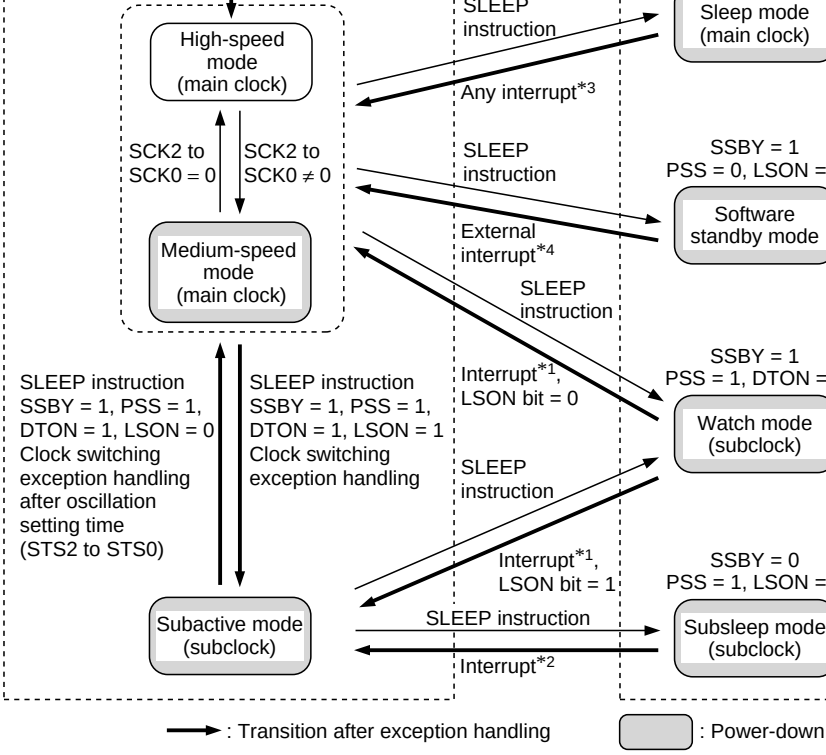
Of these, 2 to 9 are power-down modes. Sleep mode and subsleep mode are CPU mod
speed mode is a CPU and bus master mode, subactive mode is a CPU, bus master, and
supporting module mode, and module stop mode is an on-chip supporting module mo
(including bus masters other than the CPU). Certain combinations of these modes can

After a reset, the MCU is in high-speed mode and module stop mode (excluding the D

Table 25.1 shows the internal chip states in each mode, and table 25.2 shows the cond
transition to the various modes. Figure 25.1 shows a mode transition diagram.

		ing	speed		ing		operation		
External interrupts	NMI	Function-ing	Function-ing	Function-ing	Function-ing	Function-ing	Function-ing	Function-ing	Functio ing
	IRQ0								
	IRQ1								
	IRQ2								
On-chip supporting module operation	DTC	Function-ing	Medium-speed	Function-ing	Function-ing/halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)	Halted (retaine
	WDT1	Function-ing	Function-ing	Function-ing	Function-ing	Subclock operation	Subclock operation	Subclock operation	Halted (retaine
	WDT0	Function-ing	Function-ing	Function-ing	Function-ing	Halted (retained)	Subclock operation	Subclock operation	Halted (retaine
	TMR0, 1	Function-ing	Function-ing	Function-ing	Function-ing/halted (retained)	Halted (retained)	Subclock operation	Subclock operation	Halted (retaine
	FRT	Function-ing	Function-ing	Function-ing	Function-ing/halted (retained)	Halted (retained)	Halted (retained)	Halted (retained)	Halted (retaine
	TMRX, Y								
	Timer connec-tion								
	IIC0								
	IIC1								
	SCI0	Function-ing	Function-ing	Function-ing	Function-ing/halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)	Halted (reset)
	SCI1								
	SCI2								
	PWM								
	PWMX								
	HIF, PS2								
	D/A								
	A/D								
	RAM	Function-ing	Function-ing	Function-ing (DTC)	Function-ing	Retained	Function-ing	Retained	Retaine
	I/O	Function-ing	Function-ing	Function-ing	Function-ing	Retained	Function-ing	Retained	Retaine

Notes: “Halted (retained)” means that internal register values are retained. The internal state is “operation suspe
“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 retained)



Notes: When a transition is made between modes by means of an interrupt, transition cannot b
on interrupt source generation alone. Ensure that interrupt handling is performed after a
the interrupt request.
From any state except hardware standby mode, a transition to the reset state occurs wh
$\overline{\text{RES}}$ goes low.
From any state, a transition to hardware standby mode occurs when $\overline{\text{STBY}}$ goes low.
When a transition is made to watch mode or subactive mode, high-speed mode must be

1. NMI, IRQ0 to IRQ2, IRQ6, IRQ7, and WDT1 interrupts
2. NMI, IRQ0 to IRQ7, and WDT0 interrupts, WDT1 interrupt, TMR0 interrupt, TMR1 in
3. All interrupts
4. NMI, IRQ0 to IRQ2, IRQ6, IRQ7

Figure 25.1 Mode Transitions

	1	0	0	*	Software standby	High-spee medium-s
	1	0	1	*	—	—
	1	1	0	0	Watch	High-spee
	1	1	1	0	Watch	Subactive
	1	1	0	1	—	—
	1	1	1	1	Subactive	—
Subactive	0	0	*	*	—	—
	0	1	0	*	—	—
	0	1	1	*	Subsleep	Subactive
	1	0	*	*	—	—
	1	1	0	0	Watch	High-spee
	1	1	1	0	Watch	Subactive
	1	1	0	1	High-speed	—
	1	1	1	1	—	—

Legend:

*: Don’t care

—: Do not set.

Low-power control register	LPWRCR	R/W	H'00	H
Timer control/status register (WDT1)	TCSR	R/W	H'00	H
Module stop control register	MSTPCRH	R/W	H'3F	H
	MSTPCRL	R/W	H'FF	H

Notes: 1. Lower 16 bits of the address.
2. Some power down state registers are assigned to the same address as oth
In this case, register selection is performed by the FLSHE bit in the serial ti
register (STCR).

25.2 Register Descriptions

25.2.1 Standby Control Register (SBYCR)

Bit	7	6	5	4	3	2	1
	SSBY	STS2	STS1	STS0	—	SCK2	SCK1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	—	R/W	R/W

SBYCR is an 8-bit readable/writable register that performs power-down mode control

SBYCR is initialized to H'00 by a reset and in hardware standby mode. It is not initial
software standby mode.

	Transition to subsleep mode after execution of SLEEP instruction in subact
1	Transition to software standby mode, subactive mode, or watch mode after of SLEEP instruction in high-speed mode or medium-speed mode
	Transition to watch mode or high-speed mode after execution of SLEEP ins subactive mode

Bits 6 to 4—Standby Timer Select 2 to 0 (STS2 to STS0): These bits select the time waits for the clock to stabilize when software standby mode, watch mode, or subactive cleared and a transition is made to high-speed mode or medium-speed mode by means specific interrupt or instruction. With crystal oscillation, refer to table 25.4 and make a according to the operating frequency so that the standby time is at least 8 ms (the oscill settling time). With an external clock, any selection can be made.

Bit 6	Bit 5	Bit 4	
STS2	**STS1**	**STS0**	**Description**
0	0	0	Standby time = 8192 states
		1	Standby time = 16384 states
	1	0	Standby time = 32768 states
		1	Standby time = 65536 states
1	0	0	Standby time = 131072 states
		1	Standby time = 262144 states
	1	0	Reserved
		1	Standby time = 16 states*

Note: * This setting must not be used in the flash memory version.

Bit 3—Reserved: This bit cannot be modified and is always read as 0.

	1	0	Medium-speed clock is ϕ/4
		1	Medium-speed clock is ϕ/8
1	0	0	Medium-speed clock is ϕ/16
		1	Medium-speed clock is ϕ/32
	1	—	—

25.2.2 Low-Power Control Register (LPWRCR)

Bit	7	6	5	4	3	2	1
	DTON	LSON	NESEL	EXCLE	—	—	—
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	—	—	—

LPWRCR is an 8-bit readable/writable register that performs power-down mode contr

LPWRCR is initialized to H'00 by a reset and in hardware standby mode. It is not initi software standby mode.

Bit 7—Direct-Transfer On Flag (DTON): Specifies whether a direct transition is ma high-speed mode, medium-speed mode, and subactive mode when making a power-do transition by executing a SLEEP instruction. The operating mode to which the transiti after SLEEP instruction execution is determined by a combination of other control bit

	or software standby mode
	When a SLEEP instruction is executed in subactive mode, a transition is ma to high-speed mode, or a transition is made to subsleep mode

Note: * When a transition is made to watch mode or subactive mode, high-speed mo be set.

Bit 6—Low-Speed On Flag (LSON): Determines the operating mode in combination control bits when making a power-down transition by executing a SLEEP instruction. A controls whether a transition is made to high-speed mode or to subactive mode when w is cleared.

Bit 6

LSON	Description
0	When a SLEEP instruction is executed in high-speed mode or medium-spe a transition is made to sleep mode, software standby mode, or watch mode
	When a SLEEP instruction is executed in subactive mode, a transition is ma watch mode, or directly to high-speed mode
	After watch mode is cleared, a transition is made to high-speed mode (I
1	When a SLEEP instruction is executed in high-speed mode a transition is m watch mode or subactive mode*
	When a SLEEP instruction is executed in subactive mode, a transition is ma subsleep mode or watch mode
	After watch mode is cleared, a transition is made to subactive mode

Note: * When a transition is made to watch mode or subactive mode, high-speed mo be set.

Bit 5—Noise Elimination Sampling Frequency Select (NESEL): Selects the frequen the subclock (ϕSUB) input from the EXCL pin is sampled with the clock (ϕ) generated system clock oscillator. When ϕ = 5 MHz or higher, clear this bit to 0.

EXCLE	Description
0	Subclock input from EXCL pin is disabled
1	Subclock input from EXCL pin is enabled

Bits 3 to 0—Reserved: These bits cannot be modified and are always read as 0.

25.2.3 Timer Control/Status Register (TCSR)

TCSR1

Bit	7	6	5	4	3	2	1
	OVF	WT/$\overline{IT}$	TME	PSS	RST/$\overline{NMI}$	CKS2	CKS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Only 0 can be written in bit 7, to clear the flag.

TCSR1 is an 8-bit readable/writable register that performs selection of the WDT1 TCI clock, mode, etc.

Only bit 4 is described here. For details of the other bits, see section 14.2.2, Timer Co Register (TCSR).

TCSR is initialized to H'00 by a reset and in hardware standby mode. It is not initializ software standby mode.

Bit 4—Prescaler Select (PSS): Selects the WDT1 TCNT input clock.

This bit also controls the operation in a power-down mode transition. The operating m which a transition is made after execution of a SLEEP instruction is determined in cor with other control bits.

1		TCNT counts ϕSUB-based prescaler (PSM) divided clock pulses
		When a SLEEP instruction is executed in high-speed mode or medium-spe a transition is made to sleep mode, watch mode*, or subactive mode*
		When a SLEEP instruction is executed in subactive mode, a transition is ma subsleep mode, watch mode, or high-speed mode

Note: * When a transition is made to watch mode or subactive mode, high-speed mo be set.

25.2.4 Module Stop Control Register (MSTPCR)

	MSTPCRH								MSTPCRL					
Bit	7	6	5	4	3	2	1	0	7	6	5	4	3	2
	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	MSTP9	MSTP8	MSTP7	MSTP6	MSTP5	MSTP4	MSTP3	MSTP2
Initial value	0	0	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

MSTPCR comprises two 8-bit readable/writable registers that perform module stop mo

MSTPCR is initialized to H'3FFF by a reset and in hardware standby mode. It is not ini software standby mode.

MSTRCRH and MSTPCRL Bits 7 to 0—Module Stop (MSTP 15 to MSTP 0): The specify module stop mode. See table 25.4 for the method of selecting on-chip supportin

MSTPCRH, MSTPCRL Bits 7 to 0

MSTP15 to MSTP0	Description	
0	Module stop mode is cleared	(Initial value of MSTP15,
1	Module stop mode is set	(Initial value of MSTP13

the bus master operating clock. For example, if ϕ/4 is selected as the operating clock,
memory is accessed in 4 states, and internal I/O registers in 8 states.

Medium-speed mode is cleared by clearing all of bits SCK2 to SCK0 to 0. A transitio
high-speed mode and medium-speed mode is cleared at the end of the current bus cyc

If a SLEEP instruction is executed when the SSBY bit in SBYCR and the LSON bit i
are cleared to 0, a transition is made to sleep mode. When sleep mode is cleared by an
medium-speed mode is restored.

If a SLEEP instruction is executed when the SSBY bit in SBYCR is set to 1, and the L
LPWRCR and the PSS bit in TCSR (WDT1) are both cleared to 0, a transition is mad
standby mode. When software standby mode is cleared by an external interrupt, mediu
mode is restored.

When the $\overline{\text{RES}}$ pin is driven low, a transition is made to the reset state, and medium-s
cleared. The same applies in the case of a reset caused by overflow of the watchdog ti

When the $\overline{\text{STBY}}$ pin is driven low, a transition is made to hardware standby mode.

Figure 25.2 shows the timing for transition to and clearance of medium-speed mode.

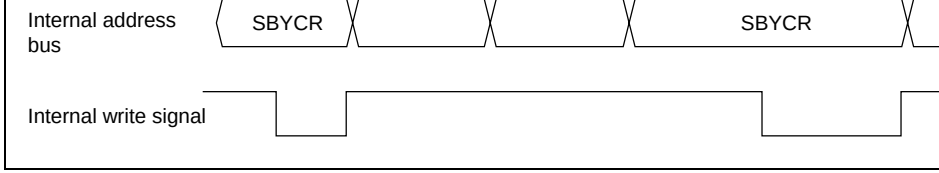


Figure 25.2 Medium-Speed Mode Transition and Clearance Timing

25.4 Sleep Mode

25.4.1 Sleep Mode

If a SLEEP instruction is executed when the SSBY bit in SBYCR and the LSON bit in are both cleared to 0, the CPU enters sleep mode. In sleep mode, CPU operation stops b contents of the CPU's internal registers are retained. Other supporting modules do not s

25.4.2 Clearing Sleep Mode

Sleep mode is cleared by any interrupt, or with the $\overline{\text{RES}}$ pin or $\overline{\text{STBY}}$ pin.

Clearing with an Interrupt: When an interrupt request signal is input, sleep mode is c interrupt exception handling is started. Sleep mode will not be cleared if interrupts are d if interrupts other than NMI have been masked by the CPU.

Clearing with the $\overline{\text{RES}}$ Pin: When the $\overline{\text{RES}}$ pin is driven low, the reset state is entered $\overline{\text{RES}}$ pin is driven high after the prescribed reset input period, the CPU begins reset exc handling.

Clearing with the $\overline{\text{STBY}}$ Pin: When the $\overline{\text{STBY}}$ pin is driven low, a transition is made standby mode.

independently.

Table 25.4 shows MSTP bits and the corresponding on-chip supporting modules.

When the corresponding MSTP bit is cleared to 0, module stop mode is cleared and th
starts operating again at the end of the bus cycle. In module stop mode, the internal sta
modules other than the SCI, A/D converter, 8-bit PWM module, and 14-bit PWM mod
retained.

After reset release, all modules other than the DTC are in module stop mode.

When an on-chip supporting module is in module stop mode, read/write access to its r
disabled.

	MSTP10	D/A converter
	MSTP9	A/D converter
	MSTP8	8-bit timers (TMRX, TMRY), timer connection
MSTPCRL	MSTP7	Serial communication interface 0 (SCI0)
	MSTP6	Serial communication interface 1 (SCI1)
	MSTP5	Serial communication interface 2 (SCI2)
	MSTP4*	I^2C bus interface (IIC) channel 0 (option)
	MSTP3*	I^2C bus interface (IIC) channel 1 (option)
	MSTP2	Host interface (HIF), keyboard matrix interrupt mask register KMIMRA), port 6 MOS pull-up control register (KMPCR), ke buffer controller (PS2)
	MSTP1*	—
	MSTP0*	—

Notes: Do not set bit 15 to 1. Bits 1 and 0 can be read or written to, but do not affect op

* Must be set to 1 in the H8S/2144 Group and H8S/2147N.

25.5.2 Usage Note

If there is conflict between DTC module stop mode setting and a DTC bus request, the has priority and the MSTP bit will not be set to 1.

Write 1 to the MSTP bit again after the DTC bus cycle.

When using the H8S/2144 Group and H8S/2147N, the MSTP bits for nonexistent modu be set to 1.

supporting modules other than the SCI, PWM, and PWMX, and of the I/O ports, are r

In this mode the oscillator stops, and therefore power dissipation is significantly reduc

25.6.2 Clearing Software Standby Mode

Software standby mode is cleared by an external interrupt (NMI pin, or pin $\overline{\text{IRQ0}}$, $\overline{\text{IRQ}}$ $\overline{\text{IRQ6}}$, or $\overline{\text{IRQ7}}$), or by means of the $\overline{\text{RES}}$ pin or $\overline{\text{STBY}}$ pin.

Clearing with an Interrupt: When an NMI, IRQ0, IRQ1, IRQ2, IRQ6, or IRQ7 inter signal is input, clock oscillation starts, and after the elapse of the time set in bits STS2 SYSCR, stable clocks are supplied to the entire chip, software standby mode is cleared interrupt exception handling is started.
Software standby mode cannot be cleared with an IRQ0, IRQ1, IRQ2, IRQ6, or IRQ7 the corresponding enable bit has been cleared to 0 or has been masked by the CPU.

Clearing with the $\overline{\text{RES}}$ Pin: When the $\overline{\text{RES}}$ pin is driven low, clock oscillation is star same time as clock oscillation starts, clocks are supplied to the entire chip. Note that t must be held low until clock oscillation stabilizes. When the $\overline{\text{RES}}$ pin goes high, the C reset exception handling.

Clearing with the $\overline{\text{STBY}}$ Pin: When the $\overline{\text{STBY}}$ pin is driven low, a transition is made standby mode.

25.6.3 Setting Oscillation Settling Time after Clearing Software Standby Mod

Bits STS2 to STS0 in SBYCR should be set as described below.

STS2	STS1	STS0	Standby Time	20 MHz	16 MHz	12 MHz	10 MHz	8 MHz	6 MHz	4 MHz
0	0	0	8192 states	0.41	0.51	0.65	0.8	1.0	1.3	2.0
		1	16384 states	0.82	1.0	1.3	1.6	2.0	2.7	4.1
	1	0	32768 states	1.6	2.0	2.7	3.3	4.1	5.5	8.2
		1	65536 states	3.3	4.1	5.5	6.6	8.2	10.9	16.4
1	0	0	131072 states	6.6	8.2	10.9	13.1	16.4	21.8	32.8
		1	262144 states	13.1	16.4	21.8	26.2	32.8	43.6	65.6
	1	0	Reserved	—	—	—	—	—	—	—
		1	16 states*	0.8	1.0	1.3	1.6	2.0	2.7	4.0

: Recommended time setting

Note: * This setting must not be used in the flash memory version.

Using an External Clock

Any value can be set. Normally, use of the minimum time is recommended.

25.6.4 Software Standby Mode Application Example

Figure 25.3 shows an example in which a transition is made to software standby mode falling edge on the NMI pin, and software standby mode is cleared at the rising edge on pin.

In this example, an NMI interrupt is accepted with the NMIEG bit in SYSCR cleared to edge specification), then the NMIEG bit is set to 1 (rising edge specification), the SSBY to 1, and a SLEEP instruction is executed, causing a transition to software standby mod

Software standby mode is then cleared at the rising edge on the NMI pin.

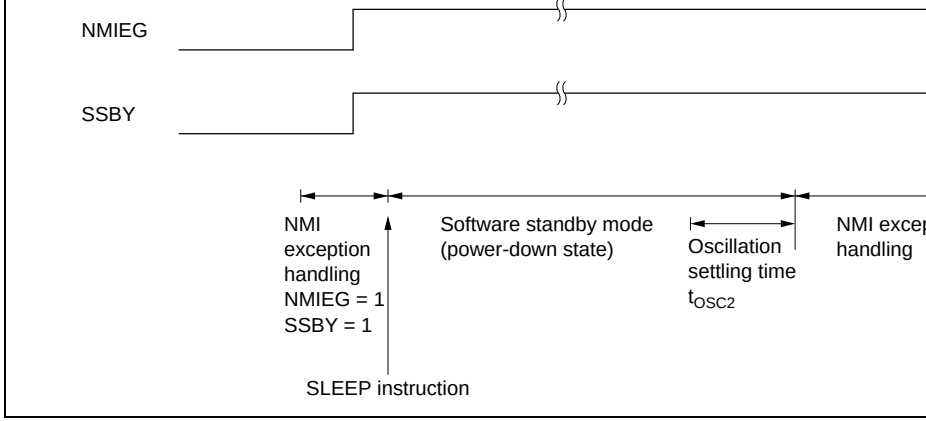


Figure 25.3 Software Standby Mode Application Example

25.6.5 Usage Note

In software standby mode, I/O port states are retained. Therefore, there is no reduction dissipation for the output current when a high-level signal is output.

Current dissipation increases while waiting for oscillation to settle.

25.7 Hardware Standby Mode

25.7.1 Hardware Standby Mode

When the $\overline{\text{STBY}}$ pin is driven low, a transition is made to hardware standby mode fro

In hardware standby mode, all functions enter the reset state and stop operation, result significant reduction in power dissipation. As long as the prescribed voltage is supplie RAM data is retained. I/O ports are set to the high-impedance state.

settling time—when using a crystal oscillator). When the $\overline{\text{RES}}$ pin is subsequently drive transition is made to the program execution state via the reset exception handling state.

25.7.2 Hardware Standby Mode Timing

Figure 25.4 shows an example of hardware standby mode timing.

When the $\overline{\text{STBY}}$ pin is driven low after the $\overline{\text{RES}}$ pin has been driven low, a transition is hardware standby mode. Hardware standby mode is cleared by driving the $\overline{\text{STBY}}$ pin h for the oscillation settling time, then changing the $\overline{\text{RES}}$ pin from low to high.

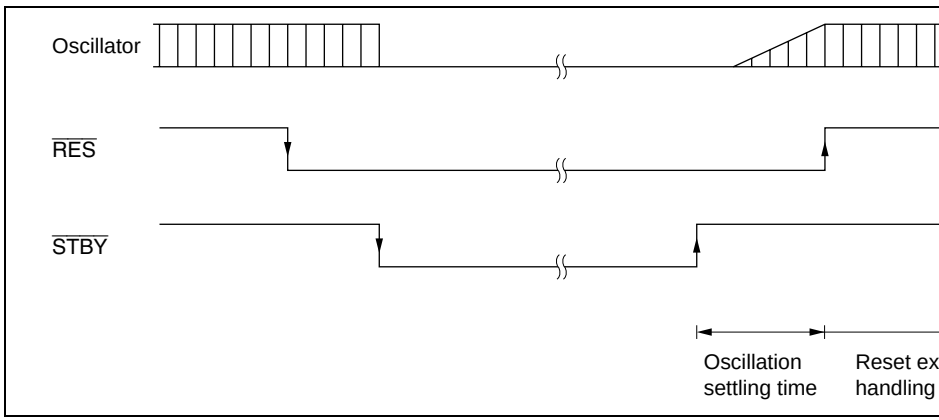


Figure 25.4 Hardware Standby Mode Timing

In this mode, the CPU and all on-chip supporting modules except WDT1 stop. As long prescribed voltage is supplied, the contents of some of the CPU's internal registers and RAM are retained, and I/O ports retain their states prior to the transition.

25.8.2 Clearing Watch Mode

Watch mode is cleared by an interrupt (WOVI1 interrupt, NMI pin, or pin $\overline{\text{IRQ0}}$, $\overline{\text{IRQ}}$ $\overline{\text{IRQ6}}$, or $\overline{\text{IRQ7}}$), or by means of the $\overline{\text{RES}}$ pin or $\overline{\text{STBY}}$ pin.

Clearing with an Interrupt: When an interrupt request signal is input, watch mode is a transition is made to high-speed mode or medium-speed mode if the LSON bit in LP cleared to 0, or to subactive mode if the LSON bit is set to 1. When making a transitio speed mode, after the elapse of the time set in bits STS2 to STS0 in SBYCR, stable cl supplied to the entire chip, and interrupt exception handling is started.

Watch mode cannot be cleared with an IRQ0, IRQ1, IRQ2, IRQ6, or IRQ7 interrupt if corresponding enable bit has been cleared to 0, or with an on-chip supporting module acceptance of the relevant interrupt has been disabled by the interrupt enable register o the CPU.

See section 25.6.3, Setting Oscillation Settling Time after Clearing Software Standby the oscillation settling time setting when making a transition from watch mode to high mode.

Clearing with the $\overline{\text{RES}}$ Pin: See "Clearing with the $\overline{\text{RES}}$ Pin" in section 25.6.2, Clear Software Standby Mode.

Clearing with the $\overline{\text{STBY}}$ Pin: When the $\overline{\text{STBY}}$ pin is driven low, a transition is made standby mode.

In this mode, the CPU and all on-chip supporting modules except TMR0, TMR1, WDT
WDT1 stop. As long as the prescribed voltage is supplied, the contents of some of the C
internal registers and on-chip RAM are retained, and I/O ports retain their states prior t
transition.

25.9.2 Clearing Subsleep Mode

Subsleep mode is cleared by an interrupt (on-chip supporting module interrupt, NMI pi
$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ7}}$), or by means of the $\overline{\text{RES}}$ pin or $\overline{\text{STBY}}$ pin.

Clearing with an Interrupt: When an interrupt request signal is input, subsleep mode
and interrupt exception handling is started. Subsleep mode cannot be cleared with an IR
IRQ7 interrupt if the corresponding enable bit has been cleared to 0, or with an on-chip
module interrupt if acceptance of the relevant interrupt has been disabled by the interru
register or masked by the CPU.

Clearing with the $\overline{\text{RES}}$ Pin: See “Clearing with the $\overline{\text{RES}}$ Pin” in section 25.6.2, Cleari
Software Standby Mode.

Clearing with the $\overline{\text{STBY}}$ Pin: When the $\overline{\text{STBY}}$ pin is driven low, a transition is made
standby mode

a transition is made to subactive mode.

In subactive mode, the CPU performs sequential program execution at low speed on th
In this mode, all on-chip supporting modules except TMR0, TMR1, WDT0, and WDT

When operating the device in subactive mode, bits SCK2 to SCK0 in SBYCR must al
to 0.

25.10.2 Clearing Subactive Mode

Subsleep mode is cleared by a SLEEP instruction, or by means of the $\overline{\text{RES}}$ pin or $\overline{\text{STB}}$

Clearing with a SLEEP Instruction: When a SLEEP instruction is executed while th
in SBYCR is set to 1, the DTON bit in LPWRCR is cleared to 0, and the PSS bit in TC
(WDT1) is set to 1, subactive mode is cleared and a transition is made to watch mode.
SLEEP instruction is executed while the SSBY bit in SBYCR is cleared to 0, the LSO
LPWRCR is set to 1, and the PSS bit in TCSR (WDT1) is set to 1, a transition is made
mode. When a SLEEP instruction is executed while the SSBY bit in SBYCR is set to
DTON bit is set to 1 and the LSON bit is cleared to 0 in LPWRCR, and the PSS bit in
(WDT1) is set to 1, a transition is made directly to high-speed mode.

Fort details of direct transition, see section 25.11, Direct Transition.

Clearing with the $\overline{\text{RES}}$ Pin: See “Clearing with the $\overline{\text{RES}}$ Pin” in section 25.6.2, Clear
Software Standby Mode.

Clearing with the $\overline{\text{STBY}}$ Pin: When the $\overline{\text{STBY}}$ pin is driven low, a transition is made
standby mode

direct transition interrupt exception handling is started.

Direct Transition from High-Speed Mode to Subactive Mode: If a SLEEP instructio executed in high-speed mode while the SSBY bit in SBYCR, the LSON bit and DTON LPWRCR, and the PSS bit in TSCR (WDT1) are all set to 1, a transition is made to sub mode.

Direct Transition from Subactive Mode to High-Speed Mode: If a SLEEP instructio executed in subactive mode while the SSBY bit in SBYCR is set to 1, the LSON bit is and the DTON bit is set to 1 in LPWRCR, and the PSS bit in TSCR (WDT1) is set to 1 elapse of the time set in bits STS2 to STS0 in SBYCR, a transition is made to directly t speed mode.

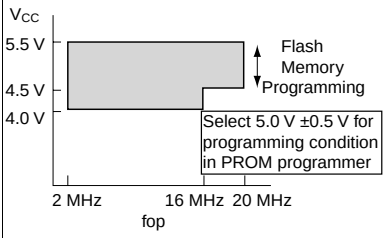
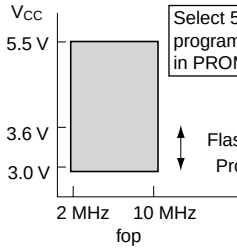
25.12 Usage Notes

1. When making a transition to subactive mode or watch mode, set the DTC to enter m mode (write 1 to the relevant bits in MSTPCR), and then read the relevant bits to co they are set to 1 before mode transition. Do not clear module stop mode (write 0 to bits in MSTPCR) until a transition from subactive mode to high-speed mode or me mode has been performed.

   If a DTC activation source occurs in sub-active mode, the DTC will be activated on module stop mode has been cleared and high-speed mode or medium-speed mode h entered.

2. The on-chip peripheral modules (DTC and TPU) which halt operation in subactive cannot clear an interrupt in subactive mode. Therefore, if a transition is made to sub mode while an interrupt is requested, the CPU interrupt source cannot be cleared. D interrupts of each on-chip peripheral module before executing a SLEEP instruction subactive mode or watch mode.

Table 26.1 Power Supply Voltage and Operating Range (1)

Product/ Power supply	5-V version	Product/ Power supply	3-V version
HD64F2148 HD64F2144 HD64F2142R		HD64F2148V HD64F2144V HD64F2142RV	
VCC1 pin VCC2 pin	V_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz) V_{cc} = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	VCC1 pin VCC2 pin	V_{cc} = 3.0 V to 5.5 V (fop = 2
VCCB pin*	$V_{cc}B$ = 5.0 V ±10% (fop = 2 to 20 MHz) $V_{cc}B$ = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	VCCB pin*	$V_{cc}B$ = 3.0 V to 5.5 V (fop = 2
AVCC pin	AV_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz) AV_{cc} = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	AVCC pin	AV_{cc} = 3.0 V to 5.5 V (fop = 2

Note: * Available only in the H8S/2148 Group.

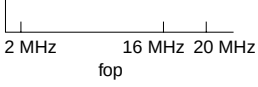
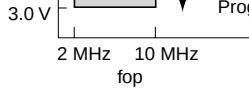
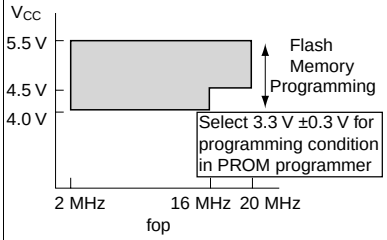
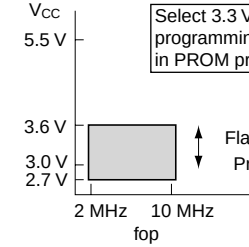
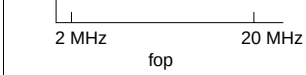
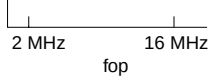
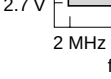
			
VCC1 pin VCC2 pin	V_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz)	VCC1 pin VCC2 pin	V_{cc} = 3.0 V to 5.5 V (fop = 2 to
VCCB pin	$V_{cc}B$ = 5.0 V ±10% (fop = 2 to 20 MHz)	VCCB pin	$V_{cc}B$ = 3.0 V to 5.5 V (fop = 2
AVCC pin	AV_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz)	AVCC pin	AV_{cc} = 3.0 V to 5.5 V (fop = 2

Table 26.1 Power Supply Voltage and Operating Range (3) (F-ZTAT A-Mask

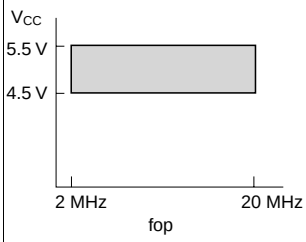
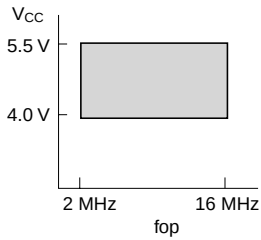
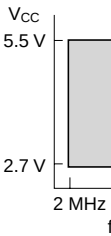
Product/ Power supply	5-V version	Product/ Power supply	3-V version
HD64F2148A HD64F2147A HD64F2144A		HD64F2148AV HD64F2147AV HD64F2144AV	
VCC1 pin	V_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz) V_{cc} = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	VCC1 pin	V_{cc} = 2.7 V to 3.6 V (fop = 2 to (CIN in use V_{cc} = 3.0 V to 3
VCL pin (VCC2)	V_{cL} = C connect	VCL pin (VCC2)	V_{cL} = V_{cc} connect
VCCB pin*	$V_{cc}B$ = 5.0 V ±10% (fop = 2 to 20 MHz) $V_{cc}B$ = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	VCCB pin*	$V_{cc}B$ = 2.7 V to 5.5 V (fop = 2 (CIN in use $V_{cc}B$ = 3.0 V to 5.5
AVCC pin	AV_{cc} = 5.0 V ±10% (fop = 2 to 20 MHz) AV_{cc} = 4.0 V to 5.5 V (fop = 2 to 16 MHz)	AVCC pin	AV_{cc} = 2.7 V to 3.6 V (fop = 2 (CIN in use AV_{cc} = 3.0 V to 3.6

Note: * Available only in the H8S/2148 Group.

HD6432148S			
VCC1 pin	V_{cc} = 5.0 V ±10%	V_{cc} = 4.0 V to 5.5 V	V_{cc} = 2.7 V to 3.6 V (CIN in use V_{cc} =
VCL pin (VCC2)	V_{cL} = C connect	V_{cL} = C connect	V_{cL} = V_{cc} connect
VCCB pin*	V_{cc}B = 5.0 V ±10%	V_{cc}B = 4.0 V to 5.5 V	V_{cc}B = 2.7 V to 5.5 (CIN in use V_{cc}B =
AVCC pin	AV_{cc} = 5.0 V ±10%	AV_{cc} = 4.0 V to 5.5 V	AV_{cc} = 2.7 V to 3.6 (CIN in use AV_{cc} =

Note: * Available only in the H8S/2148 Group.

Table 26.1 Power Supply Voltage and Operating Range (5) (Mask RO

Product/ Power supply	5-V version	4-V version	3-V ver
HD6432142			
VCC1 pin VCC2 pin	V_{cc} = 5.0 V ±10%	V_{cc} = 4.0 V to 5.5 V	V_{cc} = 2.7 V to 5.5 V
AVCC pin	AV_{cc} = 5.0 V ±10%	AV_{cc} = 4.0 V to 5.5 V	AV_{cc} = 2.7 V to 5.5

Item	Symbol	Value
Power supply voltage*	V_{CC}	–0.3 to +7.0
Input/output buffer power supply (power supply for the port A)	$V_{CC}B$	–0.3 to +7.0
Input voltage (except ports 6, 7, and A)	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 $V_{CC}B$ +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 $V_{CC}B$ +0.3 ar AV_{CC} +0.3
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 input voltage	V_{AN}	–0.3 to AV_{CC} +0.3
Operating temperature	T_{opr}	Regular specifications: –20 to +75
		Wide-range specifications: –40 to +85
Operating temperature (flash memory programming/erasing)	T_{opr}	Regular specifications: 0 to +75
		Wide-range specifications: 0 to +85
Storage temperature	T_{stg}	–55 to +125

Caution: Permanent damage to the chip may result if absolute maximum ratings are ex

Note: * Power supply voltage for VCC1 and VCC2 pins.

specifications), T_a = –40 to +85°C[*11] (wide-range specifications)

Item			Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60(KWUL = 00)[*2 *6], $\overline{KIN15}$ to $\overline{KIN8}$[*7 *8], $\overline{IRQ2}$ to $\overline{IRQ0}$[*3], $\overline{IRQ5}$ to $\overline{IRQ3}$	(1)	V_T^-	1.0	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	
			$V_T^+ - V_T^-$	0.4	—	—	
Schmitt trigger input voltage (in level switching)[*6]	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	
	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	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	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			2.0	—	AV_{CC} +0.3	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	

Output high voltage	All output pins (except P97, and P52[*4])[*5] [*8]	V_{OH}	V_{CC} –0.5 V_{CC}B –0.5	—	—	V	I_O
			3.5	—	—	V	I_O
	P97, P52[*4]		2.5	—	—	V	I_O
Output low voltage	All output pins (except $\overline{RESO}$)[*5]	V_{OL}	—	—	0.4	V	I_O
	Ports 1 to 3		—	—	1.0	V	I_O
	$\overline{RESO}$		—	—	0.4	V	I_O
Input leakage current	$\overline{RES}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V V
	$\overline{STBY}$, NMI, MD1, MD0		—	—	1.0	µA	
	Port 7		—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V V V
Input pull-up MOS current	Ports 1 to 3	$-I_P$	50	—	300	µA	V
	Ports A[*8], B, Port 6 (P6PUE = 0)		60	—	500	µA	
	Port 6 (P6PUE = 1)		15	—	150	µA	

	(4) above						
Current dissipation*9	Normal operation	I_{CC}	—	85	120	mA	
	Sleep mode		—	70	100	mA	
	Standby mode*10		—	0.01	5.0	µA	
			—	—	20.0	µA	
Analog power supply current	During A/D, D/A conversion	AI_{CC}	—	1.2	2.0	mA	
	Idle		—	0.01	5.0	µA	
Reference power supply current	During A/D conversion	AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion		—	2.0	5.0	mA	
	Idle		—	0.01	5.0	µA	
Analog power supply voltage*1		AV_{CC}	4.5	—	5.5	V	
			2.0	—	5.5	V	
RAM standby voltage		V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. In the H8S/2148 Group, P52/SCK0/SCL0 and P97/SDA0 are NMOS push-

An external pull-up resistor is necessary to provide high-level output from S SDA0 (ICE = 1).

In the H8S/2148 Group, P52/SCK0 and P97 (ICE = 0) high levels are drive

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus dri is selected is determined separately.

10. The values are for $V_{RAM} \leq V_{CC} < 4.5V$, V_{IH} min = $V_{CC} \times 0.9$, $V_{CC}B \times 0.9$, and V_{IL} 0.3 V.
11. For flash memory program/erase operations, the applicable range is T_a = 0 t (regular specifications) or T_a = 0 to +85°C (wide-range specifications).

trigger input voltage	P60(KWUL = 00)*2 *6, $\overline{KIN15}$ to $\overline{KIN8}$*7 *8, $\overline{IRQ2}$ to $\overline{IRQ0}$*3, $\overline{IRQ5}$ to $\overline{IRQ3}$		V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	V	
			$V_T^+ - V_T^-$	0.4	—	—	V	
			V_T^-	0.8	—	—	V	
			V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	V	
			$V_T^+ - V_T^-$	0.3	—	—	V	
Schmitt trigger input voltage (in level switching)*6	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V	
			V_T^+	—	—	$V_{CC} \times 0.7$		
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—		
	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	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	V_{CC} +0.3	V	
	EXTAL			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
	PA7 to PA0*7			$V_{CC}B \times 0.7$	—	$V_{CC}B + 0.3$	V	
	Port 7			2.0	—	AV_{CC} +0.3	V	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	V	

	input pins except (1) and (3) above						
Output high voltage	All output pins (except P97, and P52[*4])[*5 *8]	V_{OH}	V_{CC} –0.5 V_{CC}B –0.5	—	—	V	I_{OH}
			3.5	—	—	V	I_{OH} V_{C} 5. V_{C} to
			3.0	—	—	V	I_{OH} V_{C} V_{C}
	P97, P52[*4]		2.0	—	—	V	I_{OH}
Output low voltage	All output pins (except $\overline{\text{RESO}}$)[*5]	V_{OL}	—	—	0.4	V	I_{OL}
	Ports 1 to 3		—	—	1.0	V	I_{OL}
	$\overline{\text{RESO}}$		—	—	0.4	V	I_{OL}
Input leakage current	$\overline{\text{RES}}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V_{in} V_{C}
	$\overline{\text{STBY}}$, NMI, MD1, MD0		—	—	1.0	µA	
	Port 7		—	—	1.0	µA	V_{in} AV
Three-state leakage current (off state)	Ports 1 to 6 Ports 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} V_{C} V_{in} V_{C}

	Ports 1 to 3			30	—	200	µA	V
	Ports A[*8], B Port 6 (P6PUE = 0)			40	—	400	µA	V V
	Port 6 (P6PUE = 1)			10	—	110		
Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF	V f T
	NMI			—	—	50	pF	
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation[*9]	Normal operation		I_{CC}	—	70	100	mA	f
	Sleep mode			—	60	85	mA	f
	Standby mode[*10]			—	0.01	5.0	µA	T
				—	—	20.0		5
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	A t
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion			—	2.0	5.0	mA	
	Idle			—	0.01	5.0	µA	A t
Analog power supply voltage[*1]			AV_{CC}	4.0	—	5.5	V	C
				2.0	—	5.5	V	I
RAM standby voltage			V_{RAM}	2.0	—	—	V	

SDA0 (ICE = 1).

In the H8S/2148 Group, P52/SCK0 and P97 (ICE = 0) high levels are driven

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus driv 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 sele When a pin is in output mode, the output voltage is equivalent to the applied
7. The upper limit of the port A applied voltage is $V_{CC}B$ +0.3 V when CIN input is selected, and the lower of $V_{CC}B$ +0.3 V and AV_{CC} +0.3 V when CIN input is se When a pin is in output mode, the output voltage is equivalent to the applied
8. The port A characteristics depend on $V_{CC}B$, and the other pins characteristics on V_{CC}.
9. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, $V_{CC}B$ –0.5 V, and V_{IL} 0.5 V with all output pins unloaded and the on-chip pull-up MOSs in the off s
10. The values are for $V_{RAM} \leq V_{CC} < 4.0$ V, V_{IH} min = $V_{CC} \times 0.9$, $V_{CC}B \times 0.9$, and V_{IL} 0.3 V.
11. For flash memory program/erase operations, the applicable ranges are V_{CC} = 5.5 V and T_a = 0 to +75°C (regular specifications) or T_a = 0 to +85°C (wide-ra specifications).

voltage	00)*[5], KIN15 to KIN8*[7] *[8], IRQ2 to IRQ0*[3], IRQ5 to IRQ3		V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	V
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$ $V_{CC}B \times 0.05$	—	—	V
Schmitt trigger input voltage (in level swiching)*[6]	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V
			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	(2)	V_{IH}	$V_{CC} \times 0.9$	—	V_{CC} +0.3	V
	EXTAL			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V
	PA7 to PA0*[7]			$V_{CC}B \times 0.7$	—	$V_{CC}B$ +0.3	V
	Port 7			$V_{CC} \times 0.7$	—	AV_{CC} +0.3	V
	Input pins except (1) and (2) above			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V

	Input pins except (1) and (3) above				0.8	V	V 5
Output high voltage	All output pins (except P97, and P52[*4])[*5 *8]	V_{OH}	V_{CC} –0.5 $V_{CC}B$ –0.5	—	—	V	I_O
			V_{CC} –1.0 $V_{CC}B$ –1.0	—	—	V	I_O (V V
	P97, P52[*4]		1.0	—	—	V	I_O
Output low voltage	All output pins (except $\overline{RESO}$)[*5]	V_{OL}	—	—	0.4	V	I_O
	Ports 1 to 3		—	—	1.0	V	I_O (V I_O (4 5
	$\overline{RESO}$		—	—	0.4	V	I_O
Input leakage current	$\overline{RES}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{STBY}$, NMI, MD1, MD0		—	—	1.0	µA	V
	Port 7		—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V V V

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			—	—	15	pF
Current dissipation*9	Normal operation		I_{CC}	—	50	70	mA
	Sleep mode			—	40	60	mA
	Standby mode*10			—	0.01	5.0	µA
				—	—	20.0	µA
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA
	Idle			—	0.01	5.0	µA
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA
	During A/D, D/A conversion			—	2.0	5.0	mA
	Idle			—	0.01	5.0	µA
Analog power supply voltage*1			AV_{CC}	3.0	—	5.5	V
				2.0	—	5.5	V
RAM standby voltage			V_{RAM}	2.0	—	—	V

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.

7. The upper limit of the port A applied voltage is $V_{CC}B$ +0.3 V when CIN input is selected, and the lower of $V_{CC}B$ +0.3 V and AV_{CC} +0.3 V when CIN input is se When a pin is in output mode, the output voltage is equivalent to the applied
8. The port A characteristics depend on $V_{CC}B$, and the other pins characteristics on V_{CC}.
9. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, $V_{CC}B$ –0.5 V, and V_{IL} r 0.5 V with all output pins unloaded and the on-chip pull-up MOSs in the off s
10. The values are for $V_{RAM} \leq V_{CC} < 3.0$ V, V_{IH} min = $V_{CC} \times 0.9$, $V_{CC}B \times 0.9$, and V_{IL} 0.3 V.
11. For flash memory program/erase operations, the applicable ranges are V_{CC} = 3.6 V and Ta = 0 to +75°C.

	PA7 to PA4 (bus drive function selected)				
	Ports 1, 2, 3		—	—	10
	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1, 2, and 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

	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1, 2, and 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 26
2. When driving a Darlington pair or LED, always insert a current-limiting resist
output line, as show in figures 26.1 and 26.2.

Figure 26.1 Darlington Pair Drive Circuit (Example)

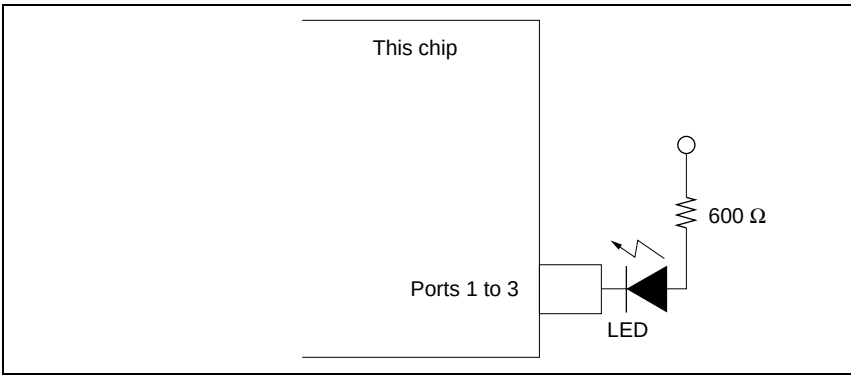


Figure 26.2 LED Drive Circuit (Example)

	$V_T - V_T$	$V_{CC} \times 0.05$	—	—		V_{CC} = 3.0 V
Input high voltage	V_{IH}	$V_{CC} \times 0.7$	—	V_{CC} +0.5	V	V_{CC} = 3.0 V
Input low voltage	V_{IL}	–0.5	—	$V_{CC} \times 0.3$		V_{CC} = 3.0 V
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 mA V_{CC} = 4.5 V
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA
Input capacitance	C_{in}	—	—	20	pF	V_{in} = 0 V, f = T_a = 25°C
Three-state leakage current (off state)	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} = 0.5 to
SCL, SDA output fall time	t_{Of}	20 + 0.1 Cb	—	250	ns	V_{CC} = 3.0 V

Conditions: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V

Applicable Pins: PS2AC, PS2AD, PS2BC, PS2BD, PS2CC, PS2CD, PA7 to PA4 (bus function selected)

Item	Symbol	Min	Typ	Max	Unit	Test Conditio
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 mA, $V_{CC}B$ = 4.5 V t
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA

26.2.3 AC Characteristics

Clock timing, control signal timing, bus timing, and timing of on-chip supporting modu following.

Figure 26.4 shows the test conditions for the AC characteristics.

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, φ = 2 MHz to maxi
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, φ = 2 MHz to
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, φ = 2 MHz to
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Clock cycle time	t_{cyc}	50	500	62.5	500	100	500	ns
Clock high pulse width	t_{CH}	17	—	20	—	30	—	ns
Clock low pulse width	t_{CL}	17	—	20	—	30	—	ns
Clock rise time	t_{Cr}	—	8	—	10	—	20	ns
Clock fall time	t_{Cf}	—	8	—	10	—	20	ns
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	10	—	20	—	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

T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, maximum operating frequency, T_a = –20 to +75°C (regular specification T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, maximum operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			Te
Item	Symbol	Min	Max	Min	Max	Min	Max	Unit	Co
$\overline{RES}$ setup time	t_{RESS}	200	—	200	—	300	—	ns	Fi
$\overline{RES}$ pulse width	t_{RESW}	20	—	20	—	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	150	—	150	—	250	—	ns	Fi
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 ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQS}	150	—	150	—	250	—	ns	
IRQ hold time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQH}	10	—	10	—	10	—	ns	
IRQ pulse width ($\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ2}$ to $\overline{IRQ0}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	200	—	ns	

T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Address delay time	t_{AD}	—	20	—	30	—	40	ns
Address setup time	t_{AS}	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	0.5 × t_{cyc} –30	—	ns
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	20	—	30	—	40	ns
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns

time 2								
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} –30	—	2.0 × t_{cyc} –40	—	2.0 × t_{cyc} –60	ns
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} –25	—	2.5 × t_{cyc} –35	—	2.5 × t_{cyc} –50	ns
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –30	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –60	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Address delay time	t_{AD}	—	30	—	45	—	60	ns
Address setup time	t_{AS}	0.5 × t_{cyc} –25	—	0.5 × t_{cyc} –35	—	0.5 × t_{cyc} –50	—	ns
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	30	—	45	—	60	ns
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns
Read data access time 1	t_{ACC1}	—	1.0 × t_{cyc} –40	—	1.0 × t_{cyc} –55	—	1.0 × t_{cyc} –80	ns
Read data access time 2	t_{ACC2}	—	1.5 × t_{cyc} –25	—	1.5 × t_{cyc} –35	—	1.5 × t_{cyc} –50	ns

time 4								
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –55	—	3.0 × t_{cyc} –80	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, φ = 32.768 kHz *, 2
maximum operating frequency, T_a = –20 to +75°C (regular specificatio
T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, φ = 32.768 kHz
maximum operating frequency, T_a = –20 to +75°C (regular specificatio
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, φ = 32.768 kHz
maximum operating frequency, T_a = –20 to +75°C

				Condition A		Condition B		Condition C		
				20 MHz		16 MHz		10 MHz		
Item			Symbol	Min	Max	Min	Max	Min	Max	Unit
I/O ports	Output data delay time		t_{PWD}	—	50	—	50	—	100	ns
	Input data setup time		t_{PRS}	30	—	30	—	50	—	
	Input data hold time		t_{PRH}	30	—	30	—	50	—	
FRT	Timer output delay time		t_{FTOD}	—	50	—	50	—	100	ns
	Timer input setup time		t_{FTIS}	30	—	30	—	50	—	
	Timer clock input setup time		t_{FTCS}	30	—	30	—	50	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{FTCWL}	2.5	—	2.5	—	2.5	—	

	setup time									
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{TMCWL}	2.5	—	2.5	—	2.5	—	
PWM, PWMX	Pulse output delay time		t_{PWOD}	—	50	—	50	—	100	ns
SCI	Input clock cycle	Asynchronous	t_{Scyc}	4	—	4	—	4	—	t_{cyc}
		Synchronous		6	—	6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6	t_{Scyc}
	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 (synchronous)		t_{TXD}	—	50	—	50	—	100	ns
	Receive data setup time (synchronous)		t_{RXS}	50	—	50	—	100	—	ns
	Receive data hold time (synchronous)		t_{RXH}	50	—	50	—	100	—	ns
A/D converter	Trigger input setup time		t_{TRGS}	30	—	30	—	50	—	ns
WDT	$\overline{\text{RESO}}$ output delay time		t_{RESD}	—	100	—	120	—	200	ns
	$\overline{\text{RESO}}$ output pulse width		t_{RESOW}	132	—	132	—	132	—	t_{cyc}

Note: * Only supporting modules that can be used in subclock operation

Condition C: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

			Condition A		Condition B		Condition C		
			20 MHz		16 MHz		10 MHz		
Item		**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
HIF read cycle	$\overline{CS}$/HA0 setup time	t_{HAR}	10	—	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time	t_{HRA}	10	—	10	—	10	—	ns
	$\overline{IOR}$ pulse width	t_{HRPW}	120	—	120	—	220	—	ns
	HDB delay time	t_{HRD}	—	100	—	100	—	200	ns
	HDB hold time	t_{HRF}	0	25	0	25	0	40	ns
	HIRQ delay time	t_{HIRQ}	—	120	—	120	—	200	ns
HIF write cycle	$\overline{CS}$/HA0 setup time	t_{HAW}	10	—	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time	t_{HWA}	10	—	10	—	10	—	ns
	$\overline{IOW}$ pulse width	t_{HWPW}	60	—	60	—	100	—	ns
	HDB setup time — Fast A20 gate not used	t_{HDW}	30	—	30	—	50	—	ns
	HDB setup time — Fast A20 gate used		45	—	55	—	85	—	ns
	HDB hold time	t_{HWD}	15	—	15	—	25	—	ns
	GA20 delay time	t_{HGA}	—	90	—	90	—	180	ns

KCLK, KD input data hold time	t_{KBIH}	150	—	—	ns
KCLK, KD input data setup time	t_{KBIS}	150	—	—	ns
KCLK, KD output delay time	t_{KBOD}	—	—	450	ns
KCLK, KD capacitive load	C_b	—	—	400	pF

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 I^2C module see section 16.4, Usage Notes.

T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3.0 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time[*5]	—	—	6.7	—	—	8.4	—	—	13.4
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10[*3] / 5[*4]	—	—	10[*3] / 5[*4]	—	—	10[*1] / 5[*2]
Nonlinearity error	—	—	±3.0	—	—	±3.0	—	—	±7.0
Offset error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Full-scale error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±4.0	—	—	±4.0	—	—	±8.0

Notes: 1. When 4.0 V ≤ AV_{CC} ≤ 5.5 V
2. When 3.0 V ≤ AV_{CC} < 4.0 V
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

$V_{SS} = AV_{SS} = 0$ V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3.0 V to AV_{CC},
$V_{SS} = AV_{SS} = 0$ V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*5	—	—	6.7	—	—	8.4	—	—	13.
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*3 5*4	—	—	10*3 5*4	—	—	10* 5*2
Nonlinearity error	—	—	±5.0	—	—	±5.0	—	—	±1
Offset error	—	—	±5.5	—	—	±5.5	—	—	±1
Full-scale error	—	—	±5.5	—	—	±5.5	—	—	±1
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.
Absolute accuracy	—	—	±6.0	—	—	±6.0	—	—	±1

Notes: 1. When 4.0 V ≤ AV_{CC} ≤ 5.5 V
2. When 3.0 V ≤ AV_{CC} < 4.0 V
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3.0 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

		Condition A			Condition B			Condition C		
		20 MHz			16 MHz			10 MHz		
Item		**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution		8	8	8	8	8	8	8	8	8
Conversion time	With 20-pF load capacitance	—	—	10	—	—	10	—	—	10
Absolute accuracy	With 2-MΩ load resistance	—	±1.0	±1.5	—	±1.0	±1.5	—	±2.0	±3
	With 4-MΩ load resistance	—	—	±1.0	—	—	±1.0	—	—	±2

Item		Symbol	Min	Typ	Max	Unit
Programming time*1 *2 *4		tP	—	10	200	ms/ 32 bytes
Erase time*1 *3 *6		tE	—	100	1200	ms/ block
Reprogramming count		N_{WEC}	100*8	10000*9	—	Times
Data retention time*10		t_{DRP}	10	—	—	Years
Programming	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after PSU-bit setting*1	y	50	—	—	µs
	Wait time after P-bit setting*1 *4	z	150	—	200	µs
	Wait time after P-bit clear*1	α	10	—	—	µs
	Wait time after PSU-bit clear*1	β	10	—	—	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	4	—	—	µs
	Maximum programming count*1 *4 *5	N	—	—	1000	Times
Erase	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after ESU-bit setting*1	y	200	—	—	µs
	Wait time after E-bit setting*1 *6	z	5	—	10	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	5	—	—	µs
	Maximum erase count*1 *6 *7	N	—	—	120	Times

Notes: 1. Set the times according to the program/erase algorithms.

2. Programming time per 32 bytes (Shows the total period for which the P-bit is set. It does not include the programming verification time.)

7. Number of times when the wait time after E-bit setting (z) = 10 ms.
   The number of erases should be set according to the actual set value of (z) t
   erasing within the maximum erase time (tE(max)).
8. Minimum number of times for which all characteristics are guaranteed after r
   (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally function
   value).
10. Data retention characteristic when rewriting is performed within the specifica
    including the minimum value.

26.2.7 Usage Note

(1) The F-ZTAT and mask ROM versions have been confirmed as fully meeting the ref
values for electrical characteristics shown in this manual. However, actual performa
figures, operating margins, noise margins, and other properties may vary due to diff
the manufacturing process, on-chip ROM, layout patterns, etc.
When system evaluation testing is carried out using the F-ZTAT version, the same
tests should also be conducted for the mask ROM version when changing over to th

(2) On-chip power supply step-down circuit
The H8S/2148 F-ZTAT does not incorporate an internal power supply step-down ci
When changing over to F-ZTAT versions or mask ROM versions incorporating an i
step-down circuit, the VCC2 pin has the same pin location as the VCL pin in a step-
circuit.
Therefore, note that the circuit patterns differ between these two types of products.

For products incorporating an internal step-down circuit, do not connect the VCL pin to the VCC power supply. (The VCC1 pin must be connected to the VCC power supply as usual.)

The power supply stabilization capacitor must be connected to the VCL pin. Use a monolithic ceramic capacitor of 0.47 µF(one or two connected in group) and locate it near the pins.

In case the power supply voltage is lower than 3.6 V, connect the capacitor in the same way as the case with no step-down circuit incorporated.

<Products incorporating internal step-down circuit>

HD6432148S, HD6432148SW,
HD6432147S, HD6432147SW
HD6432144S, HD6432143S,
HD64F2148A, HD64F2147A,
HD64F2144A

The location of the VCC2 (V_{CC} power pin in the product not incorporating an step-down circuit, is the same as the a product incorporating an internal st circuit.

It is recommended that the by-pass c are connected to the power supply te (these are reference values).

<Products not incorporating internal s circuit>

HD64F2148(V), HD64F2147N(V),
HD64F2144(V), HD64F2142(V),
HD6432142,
HD6432148SV, HD6432148SVW,
HD6432147SV, HD6432147SVW,
HD6432144SV, HD6432143SV,
HD64F2148AV, HD64F2147AV,
HD64F2144AV

Figure 26.3 Connection of External Capacitor (Mask ROM Type Incorporating Circuit and Product Not Incorporating Step-Down Circuit)

Table 26.16 Absolute Maximum Ratings

Item	Symbol	Value
Power supply voltage*1	V_{CC}	–0.3 to +7.0
Input/output buffer power supply (power supply for the port A)	$V_{CC}B$	–0.3 to +7.0
Power supply voltage*1 (3 V version)	V_{CC}	–0.3 to +4.3
Power supply voltage*2 (VCL pin)	V_{CL}	–0.3 to +4.3
Input voltage (except ports 6, 7, and A)	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 $V_{CC}B$ +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 $V_{CC}B$ +0.3 ar AV_{CC} +0.3
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)	AV_{CC}	–0.3 to +4.3
Analog input voltage	V_{AN}	–0.3 to AV_{CC} +0.3
Operating temperature	T_{opr}	Regular specifications: –20 to +75
		Wide-range specifications: –40 to +85

Notes: 1. Power supply voltage for VCC1 pin
Never exceed the maximum rating of V_{CL} in the low-power version (3-V vers
because both the VCC1 and VCL pins are connected to the V_{CC} power supp
2. It is an operating power supply voltage pin on the chip.
Never apply power supply voltage to the VCL pin in the 5- or 4-V version.
Always connect an external capacitor between the VCL pin and ground for
voltage stabilization.

specifications), T_a = –40 to +85°C (wide-range specifications)

Item			Symbol	Min	Typ	Max	Unit	T C
Schmitt trigger input voltage	P67 to P60(KWUL = 00)*2 *6, $\overline{KIN15}$ to $\overline{KIN8}$*7 *8, $\overline{IRQ2}$ to $\overline{IRQ0}$*3, $\overline{IRQ5}$ to $\overline{IRQ3}$	(1)	V_T^-	1.0	—	—	V	
			V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$		
			$V_T^+ - V_T^-$	0.4	—	—		
Schmitt trigger input voltage (in level switching)*6	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V	
			V_T^+	—	—	$V_{CC} \times 0.7$		
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—		
	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	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	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			2.0	—	AV_{CC} +0.3		
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3		

Output high voltage	All output pins (except P97, and P52[*4])[*5 *8]	V_{OH}	V_{CC} –0.5 V_{CC}B –0.5	—	—	V
			3.5	—	—	V
	P97, P52[*4]		2.0	—	—	V
Output low voltage	All output pins (except $\overline{\text{RESO}}$)[*5]	V_{OL}	—	—	0.4	V
	Ports 1 to 3		—	—	1.0	V
	$\overline{\text{RESO}}$		—	—	0.4	V
Input leakage current	$\overline{\text{RES}}$	$\lvert I_{in} \rvert$	—	—	10.0	µA
	$\overline{\text{STBY}}$, NMI, MD1, MD0		—	—	1.0	µA
	Port 7		—	—	1.0	µA
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA
Input pull-up MOS current	Ports 1 to 3	$-I_P$	30	—	300	µA
	Ports A[*8], B, Port 6 (P6PUE = 0)		60	—	600	µA
	Port 6 (P6PUE = 1)		15	—	200	µA

	(4) above						
Current dissipation*9	Normal operation	I_{CC}	—	55	70	mA	f
	Sleep mode		—	36	55	mA	f
	Standby mode*10		—	1.0	5.0	µA	T
			—	—	20.0	µA	5
Analog power supply current	During A/D, D/A conversion	AI_{CC}	—	1.2	2.0	mA	
	Idle		—	0.01	5.0	µA	A to
Reference power supply current	During A/D conversion	AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion		—	2.0	5.0	mA	
	Idle		—	0.01	5.0	µA	A to
Analog power supply voltage*1		AV_{CC}	4.5	—	5.5	V	C
			2.0	—	5.5	V	Id
RAM standby voltage		V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D conv D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in th 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V_{CC} other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. In the H8S/2148 Group, P52/SCK0/SCL0 and P97/SDA0 are NMOS push-p

An external pull-up resistor is necessary to provide high-level output from SC SDA0 (ICE = 1).

In the H8S/2148 Group, P52/SCK0 and P97 (ICE = 0) high levels are driven

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus driv is selected is determined separately.

10. The values are for $V_{RAM} \leq V_{CC} < 4.5V$, V_{IH} min = V_{CC} –0.2 V, $V_{CC}B$ –0.2 V, and 0.2 V.

trigger input voltage	P60(KWUL = 00)*2 *6, $\overline{KIN15}$ to $\overline{KIN8}$*7 *8, $\overline{IRQ2}$ to $\overline{IRQ0}$*3, $\overline{IRQ5}$ to $\overline{IRQ3}$		V_T^+	—	—	$V_{cc} \times 0.7$ $V_{cc}B \times 0.7$	V	5. V_{C} to
			$V_T^+ - V_T^-$	0.4	—	—	V	
			V_T^-	0.8	—	—	V	V_{C}
			V_T^+	—	—	$V_{cc} \times 0.7$ $V_{cc}B \times 0.7$	V	4. V_{C} to
			$V_T^+ - V_T^-$	0.3	—	—	V	
Schmitt trigger input voltage (in level switching)*6	P67 to P60 (KWUL = 01)		V_T^-	$V_{cc} \times 0.3$	—	—	V	V_{C} 5.
			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	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	V_{cc} –0.7	—	V_{cc} +0.3	V	
	EXTAL			$V_{cc} \times 0.7$	—	V_{cc} +0.3	V	
	PA7 to PA0*7			$V_{cc}B \times 0.7$	—	$V_{cc}B + 0.3$	V	
	Port 7			2.0	—	AV_{cc} +0.3	V	
	Input pins except (1) and (2) above			2.0	—	V_{cc} +0.3	V	

	NMI, EXTAL, input pins except (1) and (3) above		–0.3	—	0.8	V	
Output high voltage	All output pins (except P97, and P52[*4])[*5][*8]	V_{OH}	V_{CC} –0.5 V_{CC}B –0.5	—	—	V	I
			3.5	—	—	V	I V 5 V t
			3.0	—	—	V	I V 4 V c
	P97, P52[*4]		1.5	—	—	V	I
Output low voltage	All output pins (except $\overline{\text{RESO}}$)[*5]	V_{OL}	—	—	0.4	V	I
	Ports 1 to 3		—	—	1.0	V	I
	$\overline{\text{RESO}}$		—	—	0.4	V	I
Input leakage current	$\overline{\text{RES}}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{\text{STBY}}$, NMI, MD1, MD0		—	—	1.0	µA	V
	Port 7		—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V V V

	Ports 1 to 3			20	—	200	µA	V_{in}
	Ports A*8, B Port 6 (P6PUE = 0)			40	—	500	µA	V_{in} 4. V_{in} to
	Port 6 (P6PUE = 1)			10	—	150		
Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF	V_{in} f = T_a
	NMI			—	—	50	pF	
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation*9	Normal operation		I_{CC}	—	45	58	mA	f =
	Sleep mode			—	30	46	mA	f =
	Standby mode*10			—	1.0	5.0	µA	T_a
				—	—	20.0		50
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	AV to
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion			—	2.0	5.0	mA	
	Idle			—	0.01	5.0	µA	AV to
Analog power supply voltage*1			AV_{CC}	4.0	—	5.5	V	O
				2.0	—	5.5	V	Id
RAM standby voltage			V_{RAM}	2.0	—	—	V	

SDA0 (ICE = 1).

In the H8S/2148 Group, P52/SCK0 and P97 (ICE = 0) high levels are driven

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus dri 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 se When a pin is in output mode, the output voltage is equivalent to the applie
7. The upper limit of the port A applied voltage is $V_{CC}B$ +0.3 V when CIN input selected, and the lower of $V_{CC}B$ +0.3 V and AV_{CC} +0.3 V when CIN input is s When a pin is in output mode, the output voltage is equivalent to the applie
8. The port A characteristics depend on $V_{CC}B$, and the other pins characteristi on V_{CC}.
9. Current dissipation values are for V_{IH} min = V_{CC} –0.2 V, $V_{CC}B$ –0.2 V, and V_{IL} 0.2 V with all output pins unloaded and the on-chip pull-up MOSs in the off
10. The values are for $V_{RAM} \leq V_{CC} < 4.0$ V, V_{IH} min = V_{CC} –0.2 V, $V_{CC}B$ –0.2 V, and 0.2 V.

voltage	00) *[2], $\overline{KIN15}$ to $\overline{KIN8}$*[7] *[8], $\overline{IRQ2}$ to $\overline{IRQ0}$*[3], $\overline{IRQ5}$ to $\overline{IRQ3}$		V_T^+	—	—	$V_{cc} \times 0.7$ $V_{cc}B \times 0.7$	V
			$V_T^+ - V_T^-$	$V_{cc} \times 0.05$ $V_{cc}B \times 0.05$	—	—	V
Schmitt trigger input voltage (in level swiching)*[6]	P67 to P60 (KWUL = 01)		V_T^-	$V_{cc} \times 0.3$	—	—	V
			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	$\overline{RES}$, $\overline{STBY}$, 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	V
	PA7 to PA0*[7]			$V_{cc}B \times 0.7$	—	$V_{cc}B$ +0.3	V
	Port 7			$V_{cc} \times 0.7$	—	AV_{cc} +0.3	V
	Input pins except (1) and (2) above			$V_{cc} \times 0.7$	—	V_{cc} +0.3	V

	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, and P52[*4])[*5][*8]	V_{OH}	V_{CC} –0.5 $V_{CC}B$ –0.5	—	—	V
			V_{CC} –1.0 $V_{CC}B$ –1.0	—	—	V
	P97, P52[*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
Input leakage current	$\overline{RES}$	$\lvert I_{in} \rvert$	—	—	10.0	µA
	$\overline{STBY}$, NMI, MD1, MD0		—	—	1.0	µA
	Port 7		—	—	1.0	µA
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA

Input capacitance	RES	(4)	C_{in}	—	—	80	pF	V f T
	NMI			—	—	50	pF	
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation*9	Normal operation		I_{CC}	—	30	40	mA	f
	Sleep mode			—	20	32	mA	f
	Standby mode*10			—	1.0	5.0	µA	T
				—	—	20.0	µA	5
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	A to
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion			—	2.0	5.0	mA	
	Idle			—	0.01	5.0	µA	A to
Analog power supply voltage*1			AV_{CC}	2.7	—	3.6	V	C
				2.0	—	3.6	V	Id
RAM standby voltage			V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D conv D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in th 2.0 V to 3.6 V to AVCC and AV_{ref} pins by connection to the power supply (V_{C} other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.

7. The upper limit of the port A applied voltage is $V_{CC}B$ +0.3 V when CIN input selected, and the lower of $V_{CC}B$ +0.3 V and AV_{CC} +0.3 V when CIN input is s When a pin is in output mode, the output voltage is equivalent to the applie
8. The port A characteristics depend on $V_{CC}B$, and the other pins characteristi on V_{CC}.
9. Current dissipation values are for V_{IH} min = V_{CC} –0.2 V, $V_{CC}B$ –0.2 V, and V_{IL} 0.2 V with all output pins unloaded and the on-chip pull-up MOSs in the off
10. The values are for $V_{RAM} \leq V_{CC} < 2.7$ V, V_{IH} min = V_{CC} –0.2 V, $V_{CC}B$ –0.2 V, and 0.2 V.
11. For flash memory program/erase operations, the applicable ranges are V_{CC} 3.6 V.

	PA7 to PA4 (bus drive function selected)				
	Ports 1, 2, 3		—	—	10
	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1, 2, and 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

	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1, 2, and 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 resis output line, as show in figures 26.1 and 26.2.

	$V_T - V_T$	$V_{CC} \times 0.05$	—	—		
Input high voltage	V_{IH}	$V_{CC} \times 0.7$	—	V_{CC} +0.5	V	
Input low voltage	V_{IL}	–0.5	—	$V_{CC} \times 0.3$		
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 mA V_{CC} = 4.5 V
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA
Input capacitance	C_{in}	—	—	20	pF	V_{in} = 0 V, f = T_a = 25°C
Three-state leakage current (off state)	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} = 0.5 to
SCL, SDA output fall time	t_{Of}	20 + 0.1 Cb	—	250	ns	

Conditions: V_{CC} = 4.0 V to 5.5 V, V_{CC} = 2.7 V to 3.6 V (3 V version), V_{CC}B = 2.7 V to
V_{SS} = 0 V

Applicable Pins: PS2AC, PS2AD, PS2BC, PS2BD, PS2CC, PS2CD, PA7 to PA4 (bus function selected)

Item	Symbol	Min	Typ	Max	Unit	Test Conditio
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 mA, V_{CC}B = 4.5 V t
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA

26.3.3 AC Characteristics

Clock timing, control signal timing, bus timing, and timing of on-chip supporting modu following.

Figure 26.4 shows the test conditions for the AC characteristics.

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, ϕ = 2 MHz to maxi
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, $V_{CC}B$ = 2.7 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Clock cycle time	t_{cyc}	50	500	62.5	500	100	500	ns
Clock high pulse width	t_{CH}	17	—	20	—	30	—	ns
Clock low pulse width	t_{CL}	17	—	20	—	30	—	ns
Clock rise time	t_{Cr}	—	8	—	10	—	20	ns
Clock fall time	t_{Cf}	—	8	—	10	—	20	ns
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	10	—	20	—	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

T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, maximum operating frequency, T_a = –20 to +75°C (regular specification T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, $V_{CC}B$ = 2.7 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, maximum operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			Te
Item	Symbol	Min	Max	Min	Max	Min	Max	Unit	Co
$\overline{RES}$ setup time	t_{RESS}	200	—	200	—	300	—	ns	Fi
$\overline{RES}$ pulse width	t_{RESW}	20	—	20	—	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	150	—	150	—	250	—	ns	Fi
NMI hold time (NMI)	t_{NMIH}	10	—	10	—	10	—		
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	200	—	200	—	ns	
IRQ setup time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQS}	150	—	150	—	250	—	ns	
IRQ hold time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQH}	10	—	10	—	10	—	ns	
IRQ pulse width ($\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ2}$ to $\overline{IRQ0}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	200	—	ns	

T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, $V_{CC}B$ = 2.7 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Address delay time	t_{AD}	—	20	—	30	—	40	ns
Address setup time	t_{AS}	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	0.5 × t_{cyc} –30	—	ns
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	20	—	30	—	40	ns
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns

time 2								
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} –30	—	2.0 × t_{cyc} –40	—	2.0 × t_{cyc} –60	ns
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} –25	—	2.5 × t_{cyc} –35	—	2.5 × t_{cyc} –50	ns
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –30	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –60	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

Condition C: V_{CC} = 2.7 V to 3.6 V, $V_{CC}B$ = 2.7 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Address delay time	t_{AD}	—	30	—	45	—	60	ns
Address setup time	t_{AS}	0.5 × t_{cyc} –25	—	0.5 × t_{cyc} –35	—	0.5 × t_{cyc} –50	—	ns
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	30	—	45	—	60	ns
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns
Read data access time 1	t_{ACC1}	—	1.0 × t_{cyc} –40	—	1.0 × t_{cyc} –55	—	1.0 × t_{cyc} –80	ns
Read data access time 2	t_{ACC2}	—	1.5 × t_{cyc} –25	—	1.5 × t_{cyc} –35	—	1.5 × t_{cyc} –50	ns

time 4								
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –55	—	3.0 × t_{cyc} –80	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, φ = 32.768 kHz*, 2
maximum operating frequency, T_a = –20 to +75°C (regular specificatio
T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, $V_{CC}B$ = 4.0 V to 5.5 V, V_{SS} = 0 V, φ = 32.768 kHz
maximum operating frequency, T_a = –20 to +75°C (regular specificatio
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, $V_{CC}B$ = 2.7 V to 5.5 V, V_{SS} = 0 V, φ = 32.768 kHz
maximum operating frequency, T_a = –20 to +75°C

				Condition A		Condition B		Condition C		
				20 MHz		16 MHz		10 MHz		
Item			**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
I/O ports	Output data delay time		t_{PWD}	—	50	—	50	—	100	ns
	Input data setup time		t_{PRS}	30	—	30	—	50	—	
	Input data hold time		t_{PRH}	30	—	30	—	50	—	
FRT	Timer output delay time		t_{FTOD}	—	50	—	50	—	100	ns
	Timer input setup time		t_{FTIS}	30	—	30	—	50	—	
	Timer clock input setup time		t_{FTCS}	30	—	30	—	50	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{FTCWL}	2.5	—	2.5	—	2.5	—	

	setup time									
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{TMCWL}	2.5	—	2.5	—	2.5	—	
PWM, PWMX	Pulse output delay time		t_{PWOD}	—	50	—	50	—	100	ns
SCI	Input clock cycle	Asynchronous	t_{Scyc}	4	—	4	—	4	—	t_{cyc}
		Synchronous		6	—	6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6	t_{Scyc}
	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 (synchronous)		t_{TXD}	—	50	—	50	—	100	ns
	Receive data setup time (synchronous)		t_{RXS}	50	—	50	—	100	—	ns
	Receive data hold time (synchronous)		t_{RXH}	50	—	50	—	100	—	ns
A/D converter	Trigger input setup time		t_{TRGS}	30	—	30	—	50	—	ns
WDT	$\overline{\text{RESO}}$ output delay time		t_{RESD}	—	100	—	120	—	200	ns
	$\overline{\text{RESO}}$ output pulse width		t_{RESOW}	132	—	132	—	132	—	t_{cyc}

Note: * Only supporting modules that can be used in subclock operation

Condition C: V_{CC} = 2.7 V to 3.6 V, V_{CC}B = 2.7 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

			Condition A		Condition B		Condition C		
			20 MHz		16 MHz		10 MHz		
Item		**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
HIF read cycle	$\overline{CS}$/HA0 setup time	t_{HAR}	10	—	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time	t_{HRA}	10	—	10	—	10	—	ns
	$\overline{IOR}$ pulse width	t_{HRPW}	120	—	120	—	220	—	ns
	HDB delay time	t_{HRD}	—	100	—	100	—	200	ns
	HDB hold time	t_{HRF}	0	25	0	25	0	40	ns
	HIRQ delay time	t_{HIRQ}	—	120	—	120	—	200	ns
HIF write cycle	$\overline{CS}$/HA0 setup time	t_{HAW}	10	—	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time	t_{HWA}	10	—	10	—	10	—	ns
	$\overline{IOW}$ pulse width	t_{HWPW}	60	—	60	—	100	—	ns
	HDB setup time — Fast A20 gate not used	t_{HDW}	30	—	30	—	50	—	ns
	HDB setup time — Fast A20 gate used		45	—	55	—	85	—	ns
	HDB hold time	t_{HWD}	15	—	15	—	25	—	ns
	GA20 delay time	t_{HGA}	—	90	—	90	—	180	ns

KCLK, KD input data hold time	t_{KBIH}	150	—	—	ns
KCLK, KD input data setup time	t_{KBIS}	150	—	—	ns
KCLK, KD output delay time	t_{KBOD}	—	—	450	ns
KCLK, KD capacitive load	C_b	—	—	400	pF

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 output fall time	t_{of}	20 + 0.1 Cb	—	250	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 I^2C module see section 16.4, Usage Notes.

T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*3	—	—	6.7	—	—	8.4	—	—	13.4
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*1 5*2	—	—	10*1 5*2	—	—	5
Nonlinearity error	—	—	±3.0	—	—	±3.0	—	—	±7.0
Offset error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Full-scale error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±4.0	—	—	±4.0	—	—	±8.0

Notes: 1. When conversion time ≥ 11. 17 µs (CKS = 1 and $\phi \leq$ 12 MHz, or CKS = 0)
2. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
3. In single mode and ϕ = maximum operating frequency.

V_{SS} = AV_{SS} = 0 V, φ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 3.6 V*4, AV_{CC} = 3.0 V to 3.6 V*4, AV_{ref} = 3.0 V to AV_{CC}*
$V_{CC}B$ = 3.0 V to 5.5 V*4, V_{SS} = AV_{SS} = 0 V,
φ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*3	—	—	6.7	—	—	8.4	—	—	13.
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*1 / 5*2	—	—	10*1 / 5*2	—	—	5
Nonlinearity error	—	—	±5.0	—	—	±5.0	—	—	±11
Offset error	—	—	±5.5	—	—	±5.5	—	—	±11
Full-scale error	—	—	±5.5	—	—	±5.5	—	—	±11
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±6.0	—	—	±6.0	—	—	±12

Notes: 1. When conversion time ≥ 11. 17 µs (CKS = 1 and φ ≤ 12 MHz, or CKS = 0)
2. When conversion time < 11. 17 µs (CKS = 1 and φ > 12 MHz)
3. In single mode and φ = maximum operating frequency.
4. When using CIN, the applicable range is V_{CC} = 3.0 V to 3.6 V, AV_{CC} = 3.0 V
AVref = 3.0 V to 3.6 V, and $V_{CC}B$ = 3.0 V to 5.5 V.

T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

		Condition A			Condition B			Condition C		
		20 MHz			16 MHz			10 MHz		
Item		Min	Typ	Max	Min	Typ	Max	Min	Typ	Ma
Resolution		8	8	8	8	8	8	8	8	8
Conversion time	With 20-pF load capacitance	—	—	10	—	—	10	—	—	10
Absolute accuracy	With 2-MΩ load resistance	—	±1.0	±1.5	—	±1.0	±1.5	—	±2.0	±3.
	With 4-MΩ load resistance	—	—	±1.0	—	—	±1.0	—	—	±2.

26.3.6 Flash Memory Characteristics

Table 26.29 shows the flash memory characteristics.

						128 bytes
Erase time*1 *3 *6		tE	—	100	1200	ms/ block
Reprogramming count		N_{WEC}	100*8	10000*9	—	Times
Data retention time*10		t_{DRP}	10	—	—	Years
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	—	—	µs
	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 H'FF 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

z2, z3) to allow programming within the maximum programming time (tP(ma
wait time after P-bit setting (z1, z2, z3) must be changed with the value of th
of writing times (n) as follows.

The number of times for writing n

$1 \leq n \leq 6$ z1 = 30 µs, z3 = 10 µs

$7 \leq n \leq 1000$ z2 = 200 µs

6. Maximum erase time (tE (max))
   tE(max) = Waiting time after E-bit setting (z) × Maximum erase count (N)
7. Maximum erase count (N) should be set according to the actual setting (z) to
   erase within the maximum erase time (tE(max)).
8. Minimum number of times for which all characteristics are guaranteed after r
   (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally function
   value).
10. Data retention characteristic when rewriting is performed within the specifica
    including the minimum value.

(2) On-chip power supply step-down circuit

The following products incorporate an internal power supply step-down circuit, wh automatically drops down the internal power supply voltage to the optimum intern level: the F-ZTAT A-mask versions of the H8S/2148, H8S/2147, and H8S/2144 (HD64F2148A, HD64F2147A, and HD64F2144A) and the mask ROM versions o H8S/2148, H8S/2147, H8S/2144, and H8S/2143 (HD6432148S, HD6432148SW, HD6432147S, HD6432147SW, HD6432144S, and HD6432143S).

The voltage-stabilization capacitor (0.47 μF one or two connected in group) must b between the VCL (internal power supply step-down) and VSS pins.

Figure 26.3 shows the connection of the external capacitors.

For the 5- or 4-V version whose power supply (V_{CC}) voltage exceeds 3.6 V, do not VCL pin in a product incorporating an internal step-down circuit to the V_{CC} power (Connect the VCC1 pin to the V_{CC} power supply as usual.)

For the 3-V version whose power supply (V_{CC}) voltage is 3.6 V or lower, connect b VCL and VCC1 pins to the system power supply.

When changing from the F-ZTAT versions not incorporating an internal step-down the F-ZTAT A-mask versions or mask ROM versions incorporating an internal ste circuit, the VCL pin has the same pin location as the VCC2 pin. Therefore, note th patterns differ between these two types of products.

Item	Symbol	Value
Power supply voltage*	V_{CC}	–0.3 to +7.0
Input/output buffer power supply (power supply for the port A)	$V_{CC}B$	–0.3 to +7.0
Input voltage (except ports 6, 7, and A)	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 $V_{CC}B$ +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 $V_{CC}B$ +0.3 ar AV_{CC} +0.3
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 input voltage	V_{AN}	–0.3 to AV_{CC} +0.3
Operating temperature	T_{opr}	Regular specifications: –20 to +75
Operating temperature (flash memory programming/erasing)	T_{opr}	Regular specifications: 0 to +75
Storage temperature	T_{stg}	–55 to +125

Caution: Permanent damage to the chip may result if absolute maximum ratings are ex

Note: * Power supply voltage for VCC1 and VCC2 pins.

Item			Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60(KWUL = 00)*2 *6, $\overline{KIN15}$ to $\overline{KIN8}$*7 *8, $\overline{IRQ2}$ to $\overline{IRQ0}$*3, $\overline{IRQ5}$ to $\overline{IRQ3}$	(1)	V_T^-	1.0	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	
			$V_T^+ - V_T^-$	0.4	—	—	
Schmitt trigger input voltage (in level switching)*6	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$	
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	
	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	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	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			2.0	—	AV_{CC} +0.3	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	

Output high voltage	All output pins (except P97, and P52[*4])[*5][*8]	V_{OH}	V_{CC} –0.5 V_{CC}B –0.5	—	—	V	I_O
			3.5	—	—	V	I_O
	P97, P52[*4]		2.5	—	—	V	I_O
Output low voltage	All output pins (except $\overline{RESO}$)[*5]	V_{OL}	—	—	0.4	V	I_O
	Ports 1 to 3		—	—	1.0	V	I_O
	$\overline{RESO}$		—	—	0.4	V	I_O
Input leakage current	$\overline{RES}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{STBY}$, NMI, MD1, MD0		—	—	1.0	µA	V
	Port 7		—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A[*8], B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V V V
Input pull-up MOS current	Ports 1 to 3	$-I_P$	50	—	300	µA	V
	Ports A[*8], B, Port 6 (P6PUE = 0)		60	—	500	µA	
	Port 6 (P6PUE = 1)		15	—	150	µA	

	(4) above						
Current dissipation*9	Normal operation	I_{CC}	—	75	100	mA	
	Sleep mode		—	60	85	mA	
	Standby mode*10		—	0.01	5.0	µA	
			—	—	20.0	µA	
Analog power supply current	During A/D, D/A conversion	AI_{CC}	—	1.2	2.0	mA	
	Idle		—	0.01	5.0	µA	
Reference power supply current	During A/D conversion	AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion		—	2.0	5.0	mA	
	Idle		—	0.01	5.0	µA	
Analog power supply voltage*1		AV_{CC}	4.5	—	5.5	V	
			2.0	—	5.5	V	
RAM standby voltage		V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply ($V_{}$ other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. In the H8S/2147N, P52/SCK0/SCL0 and P97/SDA0 are NMOS push-pull o

An external pull-up resistor is necessary to provide high-level output from S SDA0 (ICE = 1).

In the H8S/2147N, P52/SCK0 and P97 (ICE = 0) high levels are driven by N

5. When IICS = 0, ICE = 0, and KBIOE = 0. Low-level output when the bus dri is selected is determined separately.

10. The values are for $V_{RAM} \leq V_{CC} < 4.5V$, V_{IH} min = $V_{CC} \times 0.9$, $V_{CC}B \times 0.9$, and V_{IL} 0.3 V.
11. For flash memory program/erase operations, the applicable range is T_a = 0 t (regular specifications).

voltage	00)*[5], KIN15 to KIN8*[7] *[8], IRQ2 to IRQ0*[3], IRQ5 to IRQ3		V_T^+	—	—	$V_{CC} \times 0.7$ $V_{CC}B \times 0.7$	V
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$ $V_{CC}B \times 0.05$	—	—	V
Schmitt trigger input voltage (in level swiching)*[6]	P67 to P60 (KWUL = 01)		V_T^-	$V_{CC} \times 0.3$	—	—	V
			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	(2)	V_{IH}	$V_{CC} \times 0.9$	—	V_{CC} +0.3	V
	EXTAL			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V
	PA7 to PA0*[7]			$V_{CC}B \times 0.7$	—	$V_{CC}B$ +0.3	V
	Port 7			$V_{CC} \times 0.7$	—	AV_{CC} +0.3	V
	Input pins except (1) and (2) above			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V

	NMI, EXTAL, input pins except (1) and (3) above		–0.3	—	$V_{CC} \times 0.2$	V	V 4
					0.8	V	V 5
Output high voltage	All output pins (except P97, and P52*4)*5 *8	V_{OH}	V_{CC} –0.5 $V_{CC}B$ –0.5	—	—	V	I_O
			V_{CC} –1.0 $V_{CC}B$ –1.0	—	—	V	I_O (V to = 4
	P97, P52*4		1.0	—	—	V	I_O
Output low voltage	All output pins (except $\overline{RESO}$)*5	V_{OL}	—	—	0.4	V	I_O
	Ports 1 to 3		—	—	1.0	V	I_O (V I_O (4 5
	$\overline{RESO}$		—	—	0.4	V	I_O
Input leakage current	$\overline{RES}$	$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{STBY}$, NMI, MD1, MD0		—	—	1.0	µA	V
	Port 7		—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A*8, B	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V V V

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			—	—	15	pF
Current dissipation*9	Normal operation		I_{CC}	—	45	60	mA
	Sleep mode			—	35	50	mA
	Standby mode*10			—	0.01	5.0	µA
				—	—	20.0	µA
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA
	Idle			—	0.01	5.0	µA
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA
	During A/D, D/A conversion			—	2.0	5.0	mA
	Idle			—	0.01	5.0	µA
Analog power supply voltage*1			AV_{CC}	3.0	—	5.5	V
				2.0	—	5.5	V
RAM standby voltage			V_{RAM}	2.0	—	—	V

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.

7. The upper limit of the port A applied voltage is $V_{CC}B$ +0.3 V when CIN input is selected, and the lower of $V_{CC}B$ +0.3 V and AV_{CC} +0.3 V when CIN input is se When a pin is in output mode, the output voltage is equivalent to the applied
8. The port A characteristics depend on $V_{CC}B$, and the other pins characteristics on V_{CC}.
9. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, $V_{CC}B$ –0.5 V, and V_{IL} r 0.5 V with all output pins unloaded and the on-chip pull-up MOSs in the off s
10. The values are for $V_{RAM} \leq V_{CC} < 3.0$ V, V_{IH} min = $V_{CC} \times 0.9$, $V_{CC}B \times 0.9$, and V_{IL} 0.3 V.
11. For flash memory program/erase operations, the applicable ranges are V_{CC} = 3.6 V and Ta = 0 to +75°C.

	function selected)				
	Ports 1, 2, 3		—	—	10
	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1, 2, and 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

	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1, 2, and 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 26
2. When driving a Darlington pair or LED, always insert a current-limiting resist
output line, as show in figures 26.1 and 26.2.

	$V_T - V_T$	$V_{CC} \times 0.05$	—	—		V_{CC} = 3.0
Input high voltage	V_{IH}	$V_{CC} \times 0.7$	—	V_{CC} +0.5	V	V_{CC} = 3.0
Input low voltage	V_{IL}	–0.5	—	$V_{CC} \times 0.3$		V_{CC} = 3.0
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 m V_{CC} = 4.5
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA
Input capacitance	C_{in}	—	—	20	pF	V_{in} = 0 V, f T_a = 25°C
Three-state leakage current (off state)	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} = 0.5 to
SCL, SDA output fall time	t_{Of}	20 + 0.1 Cb	—	250	ns	V_{CC} = 3.0

Conditions: V_{CC} = 3.0 V to 5.5 V, V_{CC}B = 3.0 V to 5.5 V, V_{SS} = 0 V

Applicable Pins: PS2AC, PS2AD, PS2BC, PS2BD, PS2CC, PS2CD, PA7 to PA4 (bu function selected)

Item	Symbol	Min	Typ	Max	Unit	Test Conditi
Output low voltage	V_{OL}	—	—	0.8	V	I_{OL} = 16 mA, V_{CC}B = 4.5 V
		—	—	0.5		I_{OL} = 8 mA
		—	—	0.4		I_{OL} = 3 mA

26.4.3 AC Characteristics

Clock timing, control signal timing, bus timing, and timing of on-chip supporting mod following.

Figure 26.4 shows the test conditions for the AC characteristics.

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, φ = 2 MHz to maxim
operating frequency, T_a = –20 to +75°C

Condition B: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, φ = 2 MHz to m
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B			
		20 MHz		10 MHz			Te
Item	Symbol	Min	Max	Min	Max	Unit	Co
Clock cycle time	t_{cyc}	50	500	100	500	ns	Fig
Clock high pulse width	t_{CH}	17	—	30	—	ns	Fig
Clock low pulse width	t_{CL}	17	—	30	—	ns	
Clock rise time	t_{Cr}	—	8	—	20	ns	
Clock fall time	t_{Cf}	—	8	—	20	ns	
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	20	—	ms	Fig Fig
Oscillation settling time in software standby (crystal)	t_{OSC2}	8	—	8	—	ms	
External clock output stabilization delay time	t_{DEXT}	500	—	500	—	µs	

Condition B: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz maximum operating frequency, T_a = –20 to +75°C

		Condition A		Condition B			
		20 MHz		10 MHz			T
Item	**Symbol**	**Min**	**Max**	**Min**	Max	**Unit**	C
$\overline{RES}$ setup time	t_{RESS}	200	—	300	—	ns	F
$\overline{RES}$ pulse width	t_{RESW}	20	—	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	150	—	250	—	ns	F
NMI hold time (NMI)	t_{NMIH}	10	—	10	—	ns	
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	200	—	ns	
IRQ setup time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQS}	150	—	250	—	ns	
IRQ hold time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQH}	10	—	10	—	ns	
IRQ pulse width ($\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ2}$ to $\overline{IRQ0}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	ns	

Condition B: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to ma
operating frequency, T_a = –20 to +75°C

		Condition A		Condition B			
		20 MHz		10 MHz			Te
Item	Symbol	Min	Max	Min	Max	Unit	Co
Address delay time	t_{AD}	—	20	—	40	ns	Fig to fig
Address setup time	t_{AS}	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –30	—	ns	
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –20	—	ns	
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	20	—	40	ns	
$\overline{AS}$ delay time	t_{ASD}	—	30	—	60	ns	
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	60	ns	
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	60	ns	
Read data setup time	t_{RDS}	15	—	35	—	ns	
Read data hold time	t_{RDH}	0	—	0	—	ns	

time 2						
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} –30	—	2.0 × t_{cyc} –60	ns
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} –25	—	2.5 × t_{cyc} –50	ns
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –30	—	3.0 × t_{cyc} –60	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	10	—	ns

Item	Symbol	20 MHz		10 MHz		Unit	Te Co
		Min	Max	Min	Max		
Address delay time	t_{AD}	—	30	—	60	ns	Fig to fig
Address setup time	t_{AS}	0.5 × t_{cyc} –25	—	0.5 × t_{cyc} –50	—	ns	
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –20	—	ns	
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	30	—	60	ns	
$\overline{AS}$ delay time	t_{ASD}	—	30	—	60	ns	
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	60	ns	
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	60	ns	
Read data setup time	t_{RDS}	15	—	35	—	ns	
Read data hold time	t_{RDH}	0	—	0	—	ns	
Read data access time 1	t_{ACC1}	—	1.0 × t_{cyc} –40	—	1.0 × t_{cyc} –80	ns	
Read data access time 2	t_{ACC2}	—	1.5 × t_{cyc} –25	—	1.5 × t_{cyc} –50	ns	

time 4						
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –80	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	10	—	ns

Condition A: V_{CC} = 5.0 V ±10%, $V_{CC}B$ = 5.0 V ±10%, V_{SS} = 0 V, φ = 32.768 kHz*, 2 MHz maximum operating frequency, T_a = –20 to +75°C

Condition B: V_{CC} = 3.0 V to 5.5 V, $V_{CC}B$ = 3.0 V to 5.5 V, V_{SS} = 0 V, φ = 32.768 kHz*, maximum operating frequency, T_a = –20 to +75°C

				Condition A		Condition B		
				20 MHz		10 MHz		
Item			Symbol	Min	Max	Min	Max	Unit
I/O ports	Output data delay time		t_{PWD}	—	50	—	100	ns
	Input data setup time		t_{PRS}	30	—	50	—	
	Input data hold time		t_{PRH}	30	—	50	—	
FRT	Timer output delay time		t_{FTOD}	—	50	—	100	ns
	Timer input setup time		t_{FTIS}	30	—	50	—	
	Timer clock input setup time		t_{FTCS}	30	—	50	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{FTCWL}	2.5	—	2.5	—	

	setup time							
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{TMCWL}	2.5	—	2.5	—	
PWM, PWMX	Pulse output delay time		t_{PWOD}	—	50	—	100	ns
SCI	Input clock cycle	Asynchro-nous	t_{Scyc}	4	—	4	—	t_{cyc}
		Synchro-nous		6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	t_{Scyc}
	Input clock rise time		t_{SCKr}	—	1.5	—	1.5	t_{cyc}
	Input clock fall time		t_{SCKf}	—	1.5	—	1.5	
	Transmit data delay time (synchronous)		t_{TXD}	—	50	—	100	ns
	Receive data setup time (synchronous)		t_{RXS}	50	—	100	—	ns
	Receive data hold time (synchronous)		t_{RXH}	50	—	100	—	ns
A/D conver-ter	Trigger input setup time		t_{TRGS}	30	—	50	—	ns
WDT	$\overline{\text{RESO}}$ output delay time		t_{RESD}	—	100	—	200	ns
	$\overline{\text{RESO}}$ output pulse width		t_{RESOW}	132	—	132	—	t_{cyc}

Note: * Only supporting modules that can be used in subclock operation

Item			Symbol	Condition A 20 MHz Min	Condition A 20 MHz Max	Condition B 10 MHz Min	Condition B 10 MHz Max	Unit
HIF read cycle	$\overline{CS}$/HA0 setup time		t_{HAR}	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time		t_{HRA}	10	—	10	—	ns
	$\overline{IOR}$ pulse width		t_{HRPW}	120	—	220	—	ns
	HDB delay time		t_{HRD}	—	100	—	200	ns
	HDB hold time		t_{HRF}	0	25	0	40	ns
	HIRQ delay time		t_{HIRQ}	—	120	—	200	ns
HIF write cycle	$\overline{CS}$/HA0 setup time		t_{HAW}	10	—	10	—	ns
	$\overline{CS}$/HA0 hold time		t_{HWA}	10	—	10	—	ns
	$\overline{IOW}$ pulse width		t_{HWPW}	60	—	100	—	ns
	HDB setup time	Fast A20 gate not used	t_{HDW}	30	—	50	—	ns
		Fast A20 gate used		45	—	85	—	ns
	HDB hold time		t_{HWD}	15	—	25	—	ns
	GA20 delay time		t_{HGA}	—	90	—	180	ns

KCLK, KD input data hold time	t_{KBIH}	150	—	—	ns
KCLK, KD input data setup time	t_{KBIS}	150	—	—	ns
KCLK, KD output delay time	t_{KBOD}	—	—	450	ns
KCLK, KD capacitive load	C_b	—	—	400	pF

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 I^2C module. see section 16.4, Usage Notes.

T_a = –20 to +75°C

Condition B: V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3.0 V to AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

	Condition A			Condition B		
	20 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution	10	10	10	10	10	10
Conversion time*5	—	—	6.7	—	—	13.4
Analog input capacitance	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*3 5*4	—	—	10*1 5*2
Nonlinearity error	—	—	±3.0	—	—	±7.0
Offset error	—	—	±3.5	—	—	±7.5
Full-scale error	—	—	±3.5	—	—	±7.5
Quantization error	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±4.0	—	—	±8.0

Notes: 1. When 4.0 V ≤ AV_{CC} ≤ 5.5 V
2. When 3.0 V ≤ AV_{CC} < 4.0 V
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

T_a = –20 to +75°C

Item	Condition A			Condition B			U
	20 MHz			10 MHz			
	Min	Typ	Max	Min	Typ	Max	
Resolution	10	10	10	10	10	10	B
Conversion time*5	—	—	6.7	—	—	13.4	µ
Analog input capacitance	—	—	20	—	—	20	p
Permissible signal-source impedance	—	—	10*3 5*4	—	—	10*1 5*2	k
Nonlinearity error	—	—	±5.0	—	—	±11.0	L
Offset error	—	—	±5.5	—	—	±11.5	L
Full-scale error	—	—	±5.5	—	—	±11.5	L
Quantization error	—	—	±0.5	—	—	±0.5	L
Absolute accuracy	—	—	±6.0	—	—	±12.0	L

Notes: 1. When 4.0 V ≤ AV_{cc} ≤ 5.5 V
2. When 3.0 V ≤ AV_{cc} < 4.0 V
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

T_a = –40 to +85°C (wide-range specifications)

Condition B: V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3.0 V to AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

		Condition A			Condition B		
		20 MHz			10 MHz		
Item		**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution		8	8	8	8	8	8
Conversion time	With 20-pF load capacitance	—	—	10	—	—	10
Absolute accuracy	With 2-MΩ load resistance	—	±1.0	±1.5	—	±2.0	±3.0
	With 4-MΩ load resistance	—	—	±1.0	—	—	±2.0

Item		Symbol	Min	Typ	Max	Unit
Programming time*1 *2 *4		tP	—	10	200	ms/ 32 bytes
Erase time*1 *3 *6		tE	—	100	1200	ms/ block
Reprogramming count		N_{WEC}	100*8	10000*9	—	Times
Data retention time*10		t_{DRP}	10	—	—	Years
Programming	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after PSU-bit setting*1	y	50	—	—	µs
	Wait time after P-bit setting*1 *4	z	150	—	200	µs
	Wait time after P-bit clear*1	α	10	—	—	µs
	Wait time after PSU-bit clear*1	β	10	—	—	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	4	—	—	µs
	Maximum programming count*1 *4 *5	N	—	—	1000	Times
Erase	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after ESU-bit setting*1	y	200	—	—	µs
	Wait time after E-bit setting*1 *6	z	5	—	10	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	5	—	—	µs
	Maximum erase count*1 *6 *7	N	—	—	120	Times

Notes: 1. Set the times according to the program/erase algorithms.

2. Programming time per 32 bytes (Shows the total period for which the P-bit in is set. It does not include the programming verification time.)

7. Number of times when the wait time after E-bit setting (z) = 10 ms.
   The number of erases should be set according to the actual set value of (z)
   erasing within the maximum erase time (tE(max)).
8. Minimum number of times for which all characteristics are guaranteed after
   (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally functio
   value).
10. Data retention characteristic when rewriting is performed within the specific
    including the minimum value.

26.4.7 Usage Note

(1) The F-ZTAT and mask ROM versions have been confirmed as fully meeting the r
values for electrical characteristics shown in this manual. However, actual perform
figures, operating margins, noise margins, and other properties may vary due to di
the manufacturing process, on-chip ROM, layout patterns, etc.
When system evaluation testing is carried out using the F-ZTAT version, the same
tests should also be conducted for the mask ROM version when changing over to t

(2) On-chip power supply step-down circuit
The H8S/2147N F-ZTAT does not incorporate an internal power supply step-down
When changing over to F-ZTAT versions or mask ROM versions incorporating an
step-down circuit, the VCC2 pin has the same pin location as the VCL pin in a step
circuit (See figure 26.3).
Therefore, note that the circuit patterns differ between these two types of products

Item	Symbol	Value
Power supply voltage*	V_{CC}	–0.3 to +7.0
Input voltage (except ports 6, 7, and A)	V_{in}	–0.3 to V_{CC} +0.3
Input voltage (CIN input not selected for port 6 and A)	V_{in}	–0.3 to V_{CC} +0.3
Input voltage (CIN input selected for port 6 and A)	V_{in}	–0.3 V to lower of voltages V_{CC} +0.3 and AV_{CC} +0.3
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 input voltage	V_{AN}	–0.3 to AV_{CC} +0.3
Operating temperature	T_{opr}	Regular specifications: –20 to +75
		Wide-range specifications: –40 to +85
Operating temperature (flash memory programming/erasing)	T_{opr}	Regular specifications: 0 to +75
		Wide-range specifications: 0 to +85
Storage temperature	T_{stg}	–55 to +125

Caution: Permanent damage to the chip may result if absolute maximum ratings are ex

Note: * Power supply voltage for VCC1 and VCC2 pins.

T_a = –40 to +85°C[*8] (wide-range specifications)

Item			Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60[*2 *5], $\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$[*5], $\overline{\text{IRQ2}}$ to $\overline{\text{IRQ0}}$[*3], $\overline{\text{IRQ5}}$ to $\overline{\text{IRQ3}}$	(1)	V_T^-	1.0	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$	V
			$V_T^+ - V_T^-$	0.4	—	—	V
Input high voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	V_{CC} +0.3	V
	EXTAL, PA7 to PA0[*5]			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V
	Port 7			2.0	—	AV_{CC} +0.3	V
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	V
Input low voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, MD1, MD0	(3)	V_{IL}	–0.3	—	0.5	V
	PA7 to PA0			–0.3	—	1.0	V
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	0.8	V
Output high voltage	All output pins[*4]		V_{OH}	V_{CC} –0.5	—	—	V
				3.5	—	—	V
Output low voltage	All output pins (except $\overline{\text{RESO}}$)[*4]		V_{OL}	—	—	0.4	V
	Ports 1 to 3			—	—	1.0	V
	$\overline{\text{RESO}}$			—	—	0.4	V

Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A, B		$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V
Input pull-up MOS current	Ports 1 to 3		$-I_P$	50	—	300	µA	V
	Ports 6, A, B			60	—	500	µA	
Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF	V f T
	NMI			—	—	50	pF	
	P52, P97, P42, P86 PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation*6	Normal operation		I_{CC}	—	75	100	mA	f
	Sleep mode			—	60	85	mA	
	Standby mode*7			—	0.01	5.0	µA	T
				—	—	20.0	µA	5
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	A to
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA	
	During A/D, D/A conversion			—	2.0	5.0	mA	
	Idle			—	0.01	5.0	µA	A to

2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V_{CC}) or other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. When IICS = 0. Low-level output when the bus drive function is selected is separately.
5. The upper limit of the applied voltage on Ports 6 and A is V_{CC} +0.3 V when not selected, and the lower of V_{CC} +0.3 V and AV_{CC} +0.3 V when CIN input i When a pin is in output mode, the output voltage is equivalent to the applie
6. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, and V_{IL} max = 0.5 V output pins unloaded and the on-chip pull-up MOSs in the off state.
7. The values are for $V_{RAM} \leq V_{CC} < 4.5V$, V_{IH} min = $V_{CC} \times 0.9$, and V_{IL} max = 0.3
8. For flash memory program/erase operations, the applicable range is T_a = 0 (regular specifications) or T_a = 0 to +85°C (wide-range specifications).

trigger input voltage	KIN15 to KIN8*5, IRQ2 to IRQ0*3, IRQ5 to IRQ3		V_T^+	—	—	$V_{CC} \times 0.7$	V	5.
			$V_T^+ - V_T^-$	0.4	—	—	V	
			V_T^-	0.8	—	—	V	V_C 4.
			V_T^+	—	—	$V_{CC} \times 0.7$	V	
			$V_T^+ - V_T^-$	0.3	—	—	V	
Input high voltage	RES, STBY, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	V_{CC} +0.3	V	
	EXTAL, PA7 to PA0*5			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
	Port 7			2.0	—	AV_{CC} +0.3	V	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	V	
Input low voltage	RES, STBY, MD1, MD0	(3)	V_{IL}	–0.3	—	0.5	V	
	PA7 to PA0			–0.3	—	1.0	V	V_C 5.
				–0.3	—	0.8	V	V_C 4.
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	0.8	V	
Output high voltage	All output pins*4		V_{OH}	V_{CC} –0.5	—	—	V	I_{OH}
				3.5	—	—	V	I_{OH} V_C 5.
				3.0	—	—	V	I_{OH} V_C 4.

current	MD1, MD0							
	Port 7			—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A, B		$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V
Input pull-up MOS current	Ports 1 to 3		$-I_P$	50	—	300	µA	V
	Ports 6, A, B			60	—	500	µA	V 5
	Ports 1 to 3			30	—	200	µA	V
	Ports 6, A, B			40	—	400	µA	V 4
Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF	V f
	NMI			—	—	50	pF	T
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation*6	Normal operation		I_{CC}	—	65	85	mA	f
	Sleep mode			—	50	70	mA	f
	Standby mode*7			—	0.01	5.0	µA	T
				—	—	20.0	µA	5
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	A t

Analog power supply voltage	AV_{CC}	4.5	—	5.5	V	
		2.0	—	5.5	V	Id
RAM standby voltage	V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D conv D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in th 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V_{CC} other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. When IICS = 0. Low-level output when the bus drive function is selected is d separately.
5. The upper limit of the applied voltage on Ports 6 and A is V_{CC} +0.3 V when C not selected, and the lower of V_{CC} +0.3 V and AV_{CC} +0.3 V when CIN input is When a pin is in output mode, the output voltage is equivalent to the applied
6. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, and V_{IL} max = 0.5 V w output pins unloaded and the on-chip pull-up MOSs in the off state.
7. The values are for $V_{RAM} \leq V_{CC} < 4.0$ V, V_{IH} min = $V_{CC} \times 0.9$, and V_{IL} max = 0.3 V
8. For flash memory program/erase operations, the applicable ranges are V_{CC} = 5.5 V and T_a = 0 to +75°C (regular specifications) or T_a = 0 to +85°C (wide-ra specifications).

Item			Symbol	Min	Typ	Max	Unit
Schmitt trigger input voltage	P67 to P60*2 *5, $\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$*5, $\overline{\text{IRQ2}}$ to $\overline{\text{IRQ0}}$*3, $\overline{\text{IRQ5}}$ to $\overline{\text{IRQ3}}$	(1)	V_T^-	$V_{CC} \times 0.2$	—	—	V
			V_T^+	—	—	$V_{CC} \times 0.7$	V
			$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	V
Input high voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, NMI, MD1, MD0	(2)	V_{IH}	$V_{CC} \times 0.9$	—	V_{CC} +0.3	V
	EXTAL, PA7 to PA0*5			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V
	Port 7			$V_{CC} \times 0.7$	—	AV_{CC} +0.3	V
	Input pins except (1) and (2) above			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V
Input low voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, MD1, MD0	(3)	V_{IL}	–0.3	—	$V_{CC} \times 0.1$	V
	PA7 to PA0			–0.3	—	$V_{CC} \times 0.2$	V
						0.8	V
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	$V_{CC} \times 0.2$	V
						0.8	V
Output high voltage	All output pins*4		V_{OH}	V_{CC} –0.5	—	—	V
				V_{CC} –1.0	—	—	V

	$\overline{\text{RESO}}$			—	—	0.4	V	I_c
Input leakage current	$\overline{\text{RES}}$		$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{\text{STBY}}$, NMI, MD1, MD0			—	—	1.0	µA	V
	Port 7			—	—	1.0	µA	V A
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A, B		$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V V
Input pull-up MOS current	Ports 1 to 3		$-I_P$	10	—	150	µA	V
	Ports 6, A, B			30	—	250	µA	V to
Input capacitance	$\overline{\text{RES}}$	(4)	C_{in}	—	—	80	pF	V f
	NMI			—	—	50	pF	T
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	
	Input pins except (4) above			—	—	15	pF	
Current dissipation*6	Normal operation		I_{CC}	—	45	60	mA	f
	Sleep mode			—	35	50	mA	
	Standby mode*7			—	0.01	5.0	µA	T
				—	—	20.0	µA	5
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	A o

		3.0	—	5.5	V
		2.0	—	5.5	V
RAM standby voltage	V_{RAM}	2.0	—	—	V

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. When IICS = 0. Low-level output when the bus drive function is selected is separately.
5. The upper limit of the applied voltage on Ports 6 and A is V_{CC} +0.3 V when not selected, and the lower of V_{CC} +0.3 V and AV_{CC} +0.3 V when CIN input i When a pin is in output mode, the output voltage is equivalent to the applie
6. Current dissipation values are for V_{IH} min = V_{CC} –0.5 V, and V_{IL} max = 0.5 V output pins unloaded and the on-chip pull-up MOSs in the off state.
7. The values are for $V_{RAM} \leq V_{CC} < 2.7$ V (mask ROM version), and $V_{RAM} \leq V_{CC} <$ (F-ZTAT version), V_{IH} min = $V_{CC} \times 0.9$, and V_{IL} max = 0.3 V.
8. For flash memory program/erase operations, the applicable ranges are V_{CC} 3.6 V and Ta = 0 to +75°C. For the F-ZTAT versions, the test condition is V higher.

	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1, 2, and 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

Conditions: V_{CC} = 2.7 V to 5.5 V (mask ROM version), V_{CC} = 3.0 V to 5.5 V (F-ZTAT
V_{SS} = 0 V, T_a = –20 to +75°C

Item		**Symbol**	**Min**	**Typ**	**Max**
Permissible output low current (per pin)	PA7 to PA4 (bus drive function selected)	I_{OL}	—	—	10
	Ports 1, 2, 3		—	—	2
	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1, 2, and 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 26
2. When driving a Darlington pair or LED, always insert a current-limiting resist
output line, as show in figures 26.1 and 26.2.

			V_{CC} = 4.5 V to
—	—	0.5	I_{OL} = 8 mA
—	—	0.4	I_{OL} = 3 mA

26.5.3 AC Characteristics

Clock timing, control signal timing, bus timing, and timing of on-chip supporting mod
following.

Figure 26.4 shows the test conditions for the AC characteristics.

Condition A: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, φ = 2 MHz to maximum operating frequer
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-rang
specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, φ = 2 MHz to maximum operating frequ
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-rang
specifications)

Condition C: V_{CC} = 2.7 V to 5.5 V (mask ROM version), V_{CC} = 3.0 V to 5.5 V (F-ZTA
V_{SS} = 0 V, φ = 2 MHz to maximum operating frequency, T_a = –20 to +75

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			
Item	Symbol	Min	Max	Min	Max	Min	Max	Unit	T C
Clock cycle time	t_{cyc}	50	500	62.5	500	100	500	ns	F
Clock high pulse width	t_{CH}	17	—	20	—	30	—	ns	F
Clock low pulse width	t_{CL}	17	—	20	—	30	—	ns	
Clock rise time	t_{Cr}	—	8	—	10	—	20	ns	
Clock fall time	t_{Cf}	—	8	—	10	—	20	ns	
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	10	—	20	—	ms	F F
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	

range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, 2 MHz to maximum o
frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°
range specifications)

Condition C: V_{CC} = 2.7 V to 5.5 V (mask ROM version), V_{CC} = 3.0 V to 5.5 V (F-ZT
V_{SS} = 0 V, ϕ = 32.768 kHz, 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**	**T C**
$\overline{RES}$ setup time	t_{RESS}	200	—	200	—	300	—	ns	F
$\overline{RES}$ pulse width	t_{RESW}	20	—	20	—	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	150	—	150	—	250	—	ns	F
NMI hold time (NMI)	t_{NMIH}	10	—	10	—	10	—	ns	
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	200	—	200	—	ns	
IRQ setup time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQS}	150	—	150	—	250	—	ns	
IRQ hold time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQH}	10	—	10	—	10	—	ns	
IRQ pulse width ($\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ2}$ to $\overline{IRQ0}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	200	—	ns	

specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequ
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-rang
specifications)

Condition C: V_{CC} = 2.7 V to 5.5 V (mask ROM version), V_{CC} = 3.0 V to 5.5 V (F-ZTA
V_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**	**T C**
Address delay time	t_{AD}	—	20	—	30	—	40	ns	F to fi
Address setup time	t_{AS}	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	0.5 × t_{cyc} –30	—	ns	
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns	
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	20	—	30	—	40	ns	
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns	
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns	
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns	
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns	
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns	

time 2								
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} –30	—	2.0 × t_{cyc} –40	—	2.0 × t_{cyc} –60	ns
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} –25	—	2.5 × t_{cyc} –35	—	2.5 × t_{cyc} –50	ns
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –30	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –60	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C

range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz*, 2 MHz to maximum o

frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C

range specifications)

Condition C: V_{CC} = 2.7 V to 5.5 V (mask ROM version), V_{CC} = 3.0 V to 5.5 V (F-ZTA

V_{SS} = 0 V, ϕ = 32.768 kHz*, 2 MHz to maximum operating frequency,

T_a = –20 to +75°C

				Condition A		Condition B		Condition C		
				20 MHz		16 MHz		10 MHz		
Item			Symbol	Min	Max	Min	Max	Min	Max	Unit
I/O ports	Output data delay time		t_{PWD}	—	50	—	50	—	100	ns
	Input data setup time		t_{PRS}	30	—	30	—	50	—	
	Input data hold time		t_{PRH}	30	—	30	—	50	—	
FRT	Timer output delay time		t_{FTOD}	—	50	—	50	—	100	ns
	Timer input setup time		t_{FTIS}	30	—	30	—	50	—	
	Timer clock input setup time		t_{FTCS}	30	—	30	—	50	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{FTCWL}	2.5	—	2.5	—	2.5	—	

	setup time									
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{TMCWL}	2.5	—	2.5	—	2.5	—	
PWMX	Pulse output delay time		t_{PWOD}	—	50	—	50	—	100	ns
SCI	Input clock cycle	Asynchro-nous	t_{Scyc}	4	—	4	—	4	—	t_{cyc}
		Synchro-nous		6	—	6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6	t_{Scyc}
	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 (synchronous)		t_{TXD}	—	50	—	50	—	100	ns
	Receive data setup time (synchronous)		t_{RXS}	50	—	50	—	100	—	ns
	Receive data hold time (synchronous)		t_{RXH}	50	—	50	—	100	—	ns
A/D conver-ter	Trigger input setup time		t_{TRGS}	30	—	30	—	50	—	ns
WDT	$\overline{\text{RESO}}$ output delay time		t_{RESD}	—	100	—	120	—	200	ns
	$\overline{\text{RESO}}$ output pulse width		t_{RESOW}	132	—	132	—	132	—	t_{cyc}

Note: * Only supporting modules that can be used in subclock operation

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} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C (re specifications), T_a = –40 to +85°C (wide-range specifications)

Condition C (mask ROM version): V_{CC} = 2.7 V to 5.5 V, AV_{CC} = 2.7 V to 5.5 V, AV_{ref} = AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

Condition C (F-ZTAT version): V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3 AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*5	—	—	6.7	—	—	8.4	—	—	13.4
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*3 5*4	—	—	10*3 5*4	—	—	10*1 5*2
Nonlinearity error	—	—	±3.0	—	—	±3.0	—	—	±7.0
Offset error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Full-scale error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±4.0	—	—	±4.0	—	—	±8.0

Notes: 1. When 4.0 V ≤ AV_{CC} ≤ 5.5 V
2. When 2.7 V ≤ AV_{CC} < 4.0 V (mask ROM version) or when 3.0 V ≤ AV_{CC} < 4.0 version)
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

specifications), T_a = –40 to +85°C (wide-range specifications)

Condition C (mask ROM version): V_{CC} = 2.7 V to 5.5 V, AV_{CC} = 2.7 V to 5.5 V, AV_{ref}
AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

Condition C (F-ZTAT version): V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} =
AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*5	—	—	6.7	—	—	8.4	—	—	13.
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*3 5*4	—	—	10*3 5*4	—	—	10* 5*2
Nonlinearity error	—	—	±5.0	—	—	±5.0	—	—	±11
Offset error	—	—	±5.5	—	—	±5.5	—	—	±11
Full-scale error	—	—	±5.5	—	—	±5.5	—	—	±11
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±6.0	—	—	±6.0	—	—	±12

Notes: 1. When 4.0 V ≤ AV_{CC} ≤ 5.5 V
2. When 2.7 V ≤ AV_{CC} < 4.0 V (mask ROM version) or when 3.0 V ≤ AV_{CC} < 4.0
version)
3. When conversion time ≥ 11. 17 µs (CKS = 1 and ϕ ≤ 12 MHz, or CKS = 0)
4. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
5. In single mode and ϕ = maximum operating frequency.

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} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C (re specifications), T_a = –40 to +85°C (wide-range specifications)

Condition C (mask ROM version): V_{CC} = 2.7 V to 5.5 V, AV_{CC} = 2.7 V to 5.5 V, AV_{ref} = AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

Condition C (F-ZTAT version): V_{CC} = 3.0 V to 5.5 V, AV_{CC} = 3.0 V to 5.5 V, AV_{ref} = 3. AV_{CC}, V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency, T_a = –20 to +75°C

		Condition A			Condition B			Condition C		
		20 MHz			16 MHz			10 MHz		
Item		**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution		8	8	8	8	8	8	8	8	8
Conversion time	With 20 pF load capacitance	—	—	10	—	—	10	—	—	10
Absolute accuracy	With 2 MΩ load resistance	—	±1.0	±1.5	—	±1.0	±1.5	—	±2.0	±3.
	With 4 MΩ load resistance	—	—	±1.0	—	—	±1.0	—	—	±2.

Item		Symbol	Min	Typ	Max	Unit
Programming time*1 *2 *4		tP	—	10	200	ms/ 32 bytes
Erase time*1 *3 *6		tE	—	100	1200	ms/ block
Reprogramming count		N_{WEC}	100*8	10000*9	—	Times
Data retention time*10		t_{DRP}	10	—	—	Years
Programming	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after PSU-bit setting*1	y	50	—	—	µs
	Wait time after P-bit setting*1 *4	z	150	—	200	µs
	Wait time after P-bit clear*1	α	10	—	—	µs
	Wait time after PSU-bit clear*1	β	10	—	—	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	4	—	—	µs
	Maximum programming count*1 *4 *5	N	—	—	1000	Times
Erase	Wait time after SWE-bit setting*1	x	10	—	—	µs
	Wait time after ESU-bit setting*1	y	200	—	—	µs
	Wait time after E-bit setting*1 *6	z	5	—	10	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	5	—	—	µs
	Maximum erase count*1 *6 *7	N	—	—	120	Times

Notes: 1. Set the times according to the program/erase algorithms.

2. Programming time per 32 bytes (Shows the total period for which the P-bit is set. It does not include the programming verification time.)

7. Number of times when the wait time after E-bit setting (z) = 10 ms.
   The number of erases should be set according to the actual set value of (z) t
   erasing within the maximum erase time (tE (max)).
8. Minimum number of times for which all characteristics are guaranteed after r
   (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally function
   value).
10. Data retention characteristic when rewriting is performed within the specifica
    including the minimum value.

26.5.7 Usage Note

(1) The F-ZTAT and mask ROM versions have been confirmed as fully meeting the ref
values for electrical characteristics shown in this manual. However, actual performa
figures, operating margins, noise margins, and other properties may vary due to diff
the manufacturing process, on-chip ROM, layout patterns, etc.
When system evaluation testing is carried out using the F-ZTAT version, the same
tests should also be conducted for the mask ROM version when changing over to th

(2) On-chip power supply step-down circuit
The H8S/2144F-ZTAT, H8S/2142F-ZTAT, and mask ROM version of H8S/2142 d
incorporate an internal power supply step-down circuit.
When changing over to F-ZTAT versions or mask ROM versions incorporating an i
step-down circuit, the VCC2 pin has the same pin location as the VCL pin in a step-
circuit (See figure 26.3).
Therefore, note that the circuit patterns differ between these two types of products.

Item	Symbol	Value
Power supply voltage*1	V_{CC}	–0.3 to +7.0
Power supply voltage*1 (3-V version)	V_{CC}	–0.3 to +4.3
Power supply voltage*2 (VCL pin)	V_{CL}	–0.3 to +4.3
Input voltage (except ports 6, 7, and A)	V_{in}	–0.3 to V_{CC} +0.3
Input voltage (CIN input not selected for port 6 and A)	V_{in}	–0.3 to V_{CC} +0.3
Input voltage (CIN input selected for port 6 and A)	V_{in}	–0.3 V to lower of voltages V_{CC} +0. and AV_{CC} +0.3
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)	AV_{CC}	–0.3 to +4.3
Analog input voltage	V_{AN}	–0.3 to AV_{CC} +0.3
Operating temperature	T_{opr}	Regular specifications: –20 to +75
		Wide-range specifications: –40 to
Operating temperature (flash memory programming/erasing)	T_{opr}	Regular specifications: –20 to +75
		Wide-range specifications: –40 to
Storage temperature	T_{stg}	–55 to +125

Caution: 1. Permanent damage to the chip may result if absolute maximum ratings exceeded.
2. Never apply more than 7.0 V to any of the pins of the 5- or 4-V version any of the pins (except for port A) of the 3-V version.

Notes: 1. Power supply voltage for VCC1 pin.
Never exceed the maximum rating of V_{CL} in the low-power version (3-V vers because both the VCC1 and VCL pins are connected to the V_{CL} power supp
2. It is an operating power supply voltage pin on the chip.
Never apply power supply voltage to the VCL pin in the 5- or 4-V version.
Always connect an external capacitor between the VCL pin and ground for voltage stabilization.

T_a = –40 to +85°C (wide-range specifications)

Item			Symbol	Min	Typ	Max	Unit	T C
Schmitt trigger input voltage	P67 to P60*2 *5, $\overline{\text{KIN15}}$ to $\overline{\text{KIN8}}$*5, $\overline{\text{IRQ2}}$ to $\overline{\text{IRQ0}}$*3, $\overline{\text{IRQ5}}$ to $\overline{\text{IRQ3}}$	(1)	V_T^-	1.0	—	—	V	
			V_T^+	—	—	$V_{CC} \times 0.7$	V	
			$V_T^+ - V_T^-$	0.4	—	—	V	
Input high voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	V_{CC} +0.3	V	
	EXTAL, PA7 to PA0*5			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
	Port 7			2.0	—	AV_{CC} +0.3	V	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	V	
Input low voltage	$\overline{\text{RES}}$, $\overline{\text{STBY}}$, MD1, MD0	(3)	V_{IL}	–0.3	—	0.5	V	
	PA7 to PA0			–0.3	—	1.0	V	
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	0.8	V	
Output high voltage	All output pins*4		V_{OH}	V_{CC} –0.5	—	—	V	I_C
				3.5	—	—	V	I_C
Output low voltage	All output pins (except $\overline{\text{RESO}}$)*4		V_{OL}	—	—	0.4	V	I_C
	Ports 1 to 3			—	—	1.0	V	I_C
	$\overline{\text{RESO}}$			—	—	0.4	V	I_C

leakage current (off state)	9, A, B						
Input pull-up MOS current	Ports 1 to 3		$-I_P$	30	—	300	µA
	Ports 6, A, B			60	—	600	µA
Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF
	NMI			—	—	50	pF
	P52, P97, P42, P86 PA7 to PA2			—	—	20	pF
	Input pins except (4) above			—	—	15	pF
Current dissipation*6	Normal operation		I_{CC}	—	55	70	mA
	Sleep mode			—	36	55	mA
	Standby mode*7			—	1.0	5.0	µA
				—	—	20.0	µA
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA
	Idle			—	0.01	5.0	µA
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA
	During A/D, D/A conversion			—	2.0	5.0	mA
	Idle			—	0.01	5.0	µA

2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V_{CC} other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. When IICS = 0. Low-level output when the bus drive function is selected is d separately.
5. The upper limit of the applied voltage on Ports 6 and A is V_{CC} +0.3 V when C not selected, and the lower of V_{CC} +0.3 V and AV_{CC} +0.3 V when CIN input is When a pin is in output mode, the output voltage is equivalent to the applied
6. Current dissipation values are for V_{IH} min = V_{CC} –0.2 V, and V_{IL} max = 0.2 V w output pins unloaded and the on-chip pull-up MOSs in the off state.
7. The values are for $V_{RAM} \leq V_{CC} < 4.5V$, V_{IH} min = V_{CC} –0.2 V, and V_{IL} max = 0.2

trigger input voltage	KIN15 to KIN8*5, IRQ2 to IRQ0*3, IRQ5 to IRQ3		V_T^+	—	—	$V_{CC} \times 0.7$	V	5
			$V_T^+ - V_T^-$	0.4	—	—	V	
			V_T^-	0.8	—	—	V	V
			V_T^+	—	—	$V_{CC} \times 0.7$	V	
			$V_T^+ - V_T^-$	0.3	—	—	V	
Input high voltage	RES, STBY, NMI, MD1, MD0	(2)	V_{IH}	V_{CC} –0.7	—	V_{CC} +0.3	V	
	EXTAL, PA7 to PA0*5			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
	Port 7			2.0	—	AV_{CC} +0.3	V	
	Input pins except (1) and (2) above			2.0	—	V_{CC} +0.3	V	
Input low voltage	RES, STBY, MD1, MD0	(3)	V_{IL}	–0.3	—	0.5	V	
	PA7 to PA0			–0.3	—	1.0	V	V 5
				–0.3	—	0.8	V	V
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	0.8	V	
Output high voltage	All output pins*4		V_{OH}	V_{CC} –0.5	—	—	V	I
				3.5	—	—	V	I V 5
				3.0	—	—	V	I V

current	MD1, MD0							
	Port 7			—	—	1.0	µA	V_{in} AV
Three-state leakage current (off state)	Ports 1 to 6, 8, 9, A, B		$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in} V_{c}
Input pull-up MOS current	Ports 1 to 3		$-I_{P}$	30	—	300	µA	V_{in}
	Ports 6, A, B			60	—	600	µA	V_{c} 5.
	Ports 1 to 3			20	—	200	µA	V_{in}
	Ports 6, A, B			40	—	500	µA	V_{c}
Input capacitance	$\overline{\text{RES}}$	(4)	C_{in}	—	—	80	pF	V_{in}
	NMI			—	—	50	pF	f =
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF	T_{a}
	Input pins except (4) above			—	—	15	pF	
Current dissipation*6	Normal operation		I_{CC}	—	45	58	mA	f =
	Sleep mode			—	30	46	mA	f =
	Standby mode*7			—	1.0	5.0	µA	T_{a}
				—	—	20.0	µA	50
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA	
	Idle			—	0.01	5.0	µA	AV to

		2.0	—	5.5	V	I
RAM standby voltage	V_{RAM}	2.0	—	—	V	

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 5.5 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.
4. When IICS = 0. Low-level output when the bus drive function is selected is separately.
5. The upper limit of the applied voltage on Ports 6 and A is V_{CC} +0.3 V when not selected, and the lower of V_{CC} +0.3 V and AV_{CC} +0.3 V when CIN input i When a pin is in output mode, the output voltage is equivalent to the applie
6. Current dissipation values are for V_{IH} min = V_{CC} –0.2 V, and V_{IL} max = 0.2 V output pins unloaded and the on-chip pull-up MOSs in the off state.
7. The values are for $V_{RAM} \leq V_{CC} < 4.0$ V, V_{IH} min = V_{CC} –0.2 V, and V_{IL} max = 0.

voltage	$\overline{IRQ2}$ to $\overline{IRQ0}$*, $\overline{IRQ5}$ to $\overline{IRQ3}$		$V_T^+ - V_T^-$	$V_{CC} \times 0.05$	—	—	V	
Input high voltage	$\overline{RES}$, $\overline{STBY}$, NMI, MD1, MD0	(2)	V_{IH}	$V_{CC} \times 0.9$	—	V_{CC} +0.3	V	
	EXTAL, PA7 to PA0*5			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
	Port 7			$V_{CC} \times 0.7$	—	AV_{CC} +0.3	V	
	Input pins except (1) and (2) above			$V_{CC} \times 0.7$	—	V_{CC} +0.3	V	
Input low voltage	$\overline{RES}$, $\overline{STBY}$, MD1, MD0	(3)	V_{IL}	–0.3	—	$V_{CC} \times 0.1$	V	
	PA7 to PA0			–0.3	—	$V_{CC} \times 0.2$	V	
	NMI, EXTAL, input pins except (1) and (3) above			–0.3	—	$V_{CC} \times 0.2$	V	
Output high voltage	All output pins*4		V_{OH}	V_{CC} –0.5	—	—	V	I_O
				V_{CC} –1.0	—	—	V	I_O (V to
Output low voltage	All output pins (except $\overline{RESO}$)*4		V_{OL}	—	—	0.4	V	I_O
	Ports 1 to 3			—	—	1.0	V	I_O
	$\overline{RESO}$			—	—	0.4	V	I_O
Input leakage current	$\overline{RES}$		$\lvert I_{in} \rvert$	—	—	10.0	µA	V
	$\overline{STBY}$, NMI, MD1, MD0			—	—	1.0	µA	V
	Port 7			—	—	1.0	µA	V A

Input capacitance	$\overline{RES}$	(4)	C_{in}	—	—	80	pF
	NMI			—	—	50	pF
	P52, P97, P42, P86, PA7 to PA2			—	—	20	pF
	Input pins except (4) above			—	—	15	pF
Current dissipation*6	Normal operation		I_{CC}	—	30	40	mA
	Sleep mode			—	20	32	mA
	Standby mode*7			—	1.0	5.0	µA
				—	—	20.0	µA
Analog power supply current	During A/D, D/A conversion		AI_{CC}	—	1.2	2.0	mA
	Idle			—	0.01	5.0	µA
Reference power supply current	During A/D conversion		AI_{ref}	—	0.5	1.0	mA
	During A/D, D/A conversion			—	2.0	5.0	mA
	Idle			—	0.01	5.0	µA
Analog power supply voltage*1			AV_{CC}	2.7	—	3.6	V
				2.0	—	3.6	V
RAM standby voltage			V_{RAM}	2.0	—	—	V

Notes: 1. **Do not leave the AVCC, AV_{ref}, and AVSS pins open even if the A/D con D/A converter are not used.**

Even if the A/D converter and D/A converter are not used, apply a value in t 2.0 V to 3.6 V to AVCC and AV_{ref} pins by connection to the power supply (V other method. Ensure that $AV_{ref} \leq AV_{CC}$.

2. P67 to P60 include supporting module inputs multiplexed on those pins.
3. $\overline{IRQ2}$ includes the $\overline{ADTRG}$ signal multiplexed on that pin.

3.6 V.

	$\overline{\text{RESO}}$		—	—	3
	Other output pins		—	—	2
Permissible output low current (total)	Total of ports 1, 2, and 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

Conditions: V_{CC} = 2.7 V to 3.6 V, V_{SS} = 0 V, T_a = –20 to +75°C

Item		Symbol	Min	Typ	Max
Permissible output low current (per pin)	PA7 to PA4 (bus drive function selected)	I_{OL}	—	—	10
	Ports 1, 2, 3		—	—	2
	$\overline{\text{RESO}}$		—	—	1
	Other output pins		—	—	1
Permissible output low current (total)	Total of ports 1, 2, and 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 resis output line, as show in figures 26.1 and 26.2.

—	—	0.5	I_{OL} = 8 mA
—	—	0.4	I_{OL} = 3 mA

26.6.3 AC Characteristics

Clock timing, control signal timing, bus timing, and timing of on-chip supporting modu
following.

Figure 26.4 shows the test conditions for the AC characteristics.

Condition A: V_{CC} = 5.0 V ±10%, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating freque
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-ra
specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating freq
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-ra
specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating freq
T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Clock cycle time	t_{cyc}	50	500	62.5	500	100	500	ns
Clock high pulse width	t_{CH}	17	—	20	—	30	—	ns
Clock low pulse width	t_{CL}	17	—	20	—	30	—	ns
Clock rise time	t_{Cr}	—	8	—	10	—	20	ns
Clock fall time	t_{Cf}	—	8	—	10	—	20	ns
Oscillation settling time at reset (crystal)	t_{OSC1}	10	—	10	—	20	—	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

range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz, 2 MHz to maximum op frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, V_{SS} = 0 V, ϕ = 32.768 kHz, 2 MHz to maximum op frequency, T_a = –20 to +75°C

		Condition A		Condition B		Condition C			
		20 MHz		16 MHz		10 MHz			Te
Item	Symbol	Min	Max	Min	Max	Min	Max	Unit	Co
$\overline{RES}$ setup time	t_{RESS}	200	—	200	—	300	—	ns	Fi
$\overline{RES}$ pulse width	t_{RESW}	20	—	20	—	20	—	t_{cyc}	
NMI setup time (NMI)	t_{NMIS}	150	—	150	—	250	—	ns	Fi
NMI hold time (NMI)	t_{NMIH}	10	—	10	—	10	—	ns	
NMI pulse width (exiting software standby mode)	t_{NMIW}	200	—	200	—	200	—	ns	
IRQ setup time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQS}	150	—	150	—	250	—	ns	
IRQ hold time ($\overline{IRQ7}$ to $\overline{IRQ0}$)	t_{IRQH}	10	—	10	—	10	—	ns	
IRQ pulse width ($\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ2}$ to $\overline{IRQ0}$) (exiting software standby mode)	t_{IRQW}	200	—	200	—	200	—	ns	

specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating frec
T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°C (wide-ra
specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, V_{SS} = 0 V, ϕ = 2 MHz to maximum operating frec
T_a = –20 to +75°C

		Condition A		Condition B		Condition C		
		20 MHz		16 MHz		10 MHz		
Item	**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Address delay time	t_{AD}	—	20	—	30	—	40	ns
Address setup time	t_{AS}	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	0.5 × t_{cyc} –30	—	ns
Address hold time	t_{AH}	0.5 × t_{cyc} –10	—	0.5 × t_{cyc} –15	—	0.5 × t_{cyc} –20	—	ns
$\overline{CS}$ delay time ($\overline{IOS}$)	t_{CSD}	—	20	—	30	—	40	ns
$\overline{AS}$ delay time	t_{ASD}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 1	t_{RSD1}	—	30	—	45	—	60	ns
$\overline{RD}$ delay time 2	t_{RSD2}	—	30	—	45	—	60	ns
Read data setup time	t_{RDS}	15	—	20	—	35	—	ns
Read data hold time	t_{RDH}	0	—	0	—	0	—	ns

time 2								
Read data access time 3	t_{ACC3}	—	2.0 × t_{cyc} –30	—	2.0 × t_{cyc} –40	—	2.0 × t_{cyc} –60	ns
Read data access time 4	t_{ACC4}	—	2.5 × t_{cyc} –25	—	2.5 × t_{cyc} –35	—	2.5 × t_{cyc} –50	ns
Read data access time 5	t_{ACC5}	—	3.0 × t_{cyc} –30	—	3.0 × t_{cyc} –40	—	3.0 × t_{cyc} –60	ns
$\overline{WR}$ delay time 1	t_{WRD1}	—	30	—	45	—	60	ns
$\overline{WR}$ delay time 2	t_{WRD2}	—	30	—	45	—	60	ns
$\overline{WR}$ pulse width 1	t_{WSW1}	1.0 × t_{cyc} –20	—	1.0 × t_{cyc} –30	—	1.0 × t_{cyc} –40	—	ns
$\overline{WR}$ pulse width 2	t_{WSW2}	1.5 × t_{cyc} –20	—	1.5 × t_{cyc} –30	—	1.5 × t_{cyc} –40	—	ns
Write data delay time	t_{WDD}	—	30	—	45	—	60	ns
Write data setup time	t_{WDS}	0	—	0	—	0	—	ns
Write data hold time	t_{WDH}	10	—	15	—	20	—	ns
$\overline{WAIT}$ setup time	t_{WTS}	30	—	45	—	60	—	ns
$\overline{WAIT}$ hold time	t_{WTH}	5	—	5	—	10	—	ns

frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°
range specifications)

Condition B: V_{CC} = 4.0 V to 5.5 V, V_{SS} = 0 V, ϕ = 32.768 kHz*, 2 MHz to maximum
frequency, T_a = –20 to +75°C (regular specifications), T_a = –40 to +85°
range specifications)

Condition C: V_{CC} = 2.7 V to 3.6 V, V_{SS} = 0 V, ϕ = 32.768 kHz*, 2 MHz to maximum
frequency, T_a = –20 to +75°C

				Condition A		Condition B		Condition C		
				20 MHz		16 MHz		10 MHz		
Item			**Symbol**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
I/O ports	Output data delay time		t_{PWD}	—	50	—	50	—	100	ns
	Input data setup time		t_{PRS}	30	—	30	—	50	—	
	Input data hold time		t_{PRH}	30	—	30	—	50	—	
FRT	Timer output delay time		t_{FTOD}	—	50	—	50	—	100	ns
	Timer input setup time		t_{FTIS}	30	—	30	—	50	—	
	Timer clock input setup time		t_{FTCS}	30	—	30	—	50	—	
	Timer clock pulse width	Single edge	t_{FTCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{FTCWL}	2.5	—	2.5	—	2.5	—	

	setup time									
	Timer clock pulse width	Single edge	t_{TMCWH}	1.5	—	1.5	—	1.5	—	t_{cyc}
		Both edges	t_{TMCWL}	2.5	—	2.5	—	2.5	—	
PWMX	Pulse output delay time		t_{PWOD}	—	50	—	50	—	100	ns
SCI	Input clock cycle	Asynchronous	t_{Scyc}	4	—	4	—	4	—	t_{cyc}
		Synchronous		6	—	6	—	6	—	
	Input clock pulse width		t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6	t_{Scyc}
	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 (synchronous)		t_{TXD}	—	50	—	50	—	100	ns
	Receive data setup time (synchronous)		t_{RXS}	50	—	50	—	100	—	ns
	Receive data hold time (synchronous)		t_{RXH}	50	—	50	—	100	—	ns
A/D converter	Trigger input setup time		t_{TRGS}	30	—	30	—	50	—	ns
WDT	$\overline{\text{RESO}}$ output delay time		t_{RESD}	—	100	—	120	—	200	ns
	$\overline{\text{RESO}}$ output pulse width		t_{RESOW}	132	—	132	—	132	—	t_{cyc}

Note: * Only supporting modules that can be used in subclock operation

T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Ma**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*3	—	—	6.7	—	—	8.4	—	—	13.
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*1 5*2	—	—	10*1 5*2	—	—	5
Nonlinearity error	—	—	±3.0	—	—	±3.0	—	—	±7.(
Offset error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Full-scale error	—	—	±3.5	—	—	±3.5	—	—	±7.5
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±4.0	—	—	±4.0	—	—	±8.(

Notes: 1. When conversion time ≥ 11. 17 µs (CKS = 1 and $\phi \leq$ 12 MHz, or CKS = 0)
2. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
3. In single mode and ϕ = maximum operating frequency.

V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Condition C: V_{CC} = 3.0 V to 3.6 V*4, AV_{CC} = 3.0 V to 3.6 V*4, AV_{ref} = 3.0 V to AV_{CC}*4
V_{SS} = AV_{SS} = 0 V, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

	Condition A			Condition B			Condition C		
	20 MHz			16 MHz			10 MHz		
Item	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
Resolution	10	10	10	10	10	10	10	10	10
Conversion time*3	—	—	6.7	—	—	8.4	—	—	13.4
Analog input capacitance	—	—	20	—	—	20	—	—	20
Permissible signal-source impedance	—	—	10*1 / 5*2	—	—	10*1 / 5*2	—	—	5
Nonlinearity error	—	—	±5.0	—	—	±5.0	—	—	±11.0
Offset error	—	—	±5.5	—	—	±5.5	—	—	±11.5
Full-scale error	—	—	±5.5	—	—	±5.5	—	—	±11.5
Quantization error	—	—	±0.5	—	—	±0.5	—	—	±0.5
Absolute accuracy	—	—	±6.0	—	—	±6.0	—	—	±12.0

Notes: 1. When conversion time ≥ 11. 17 µs (CKS = 1 and $\phi \leq$ 12 MHz, or CKS = 0)
2. When conversion time < 11. 17 µs (CKS = 1 and ϕ > 12 MHz)
3. In single mode and ϕ = maximum operating frequency.
4. When using CIN, the applicable range is V_{CC} = 3.0 V to 3.6 V, AV_{CC} = 3.0 V to AVref = 3.0 V to 3.6 V.

T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

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, ϕ = 2 MHz to maximum operating frequency,
T_a = –20 to +75°C

		Condition A			Condition B			Condition
		20 MHz			16 MHz			10 MHz
Item		Min	Typ	Max	Min	Typ	Max	Min	Typ	M
Resolution		8	8	8	8	8	8	8	8	8
Conversion time	With 20-pF load capacitance	—	—	10	—	—	10	—	—	1
Absolute accuracy	With 2-MΩ load resistance	—	±1.0	±1.5	—	±1.0	±1.5	—	±2.0	±3
	With 4-MΩ load resistance	—	—	±1.0	—	—	±1.0	—	—	±

26.6.6 Flash Memory Characteristics

Table 26.67 shows the flash memory characteristics.

Erase time*1*3*6		tE	—	100	1200	ms/ block
Reprogramming count		N_{WEC}	100*8	10000*9	—	Times
Data retention time*10		t_{DRP}	10	—	—	Years
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	10	—	—	µs
	Wait time after PSU-bit clear*1	β	10	—	—	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	4	—	—	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	—	—	µs
	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 H'FF 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

z2, z3) to allow programming within the maximum programming time (tP (m
wait time after P-bit setting (z1, z2, z3) must be changed with the value of t
of writing times (n) as follows.
The number of times for writing n

$1 \leq n \leq 6$	z1 = 30 µs, z3 = 10 µs
$7 \leq n \leq 1000$	z2 = 200 µs

6. Maximum erase time (tE (max))
   tE (max) = Waiting time after E-bit setting (z) × Maximum erase count (N)
7. Maximum erase count (N) should be set according to the actual setting (z)
   erase within the maximum erase time (tE (max)).
8. Minimum number of times for which all characteristics are guaranteed after
   (Guarantee range is 1 to minimum value).
9. Reference value for 25°C (as a guideline, rewriting should normally functio
   value).
10. Data retention characteristic when rewriting is performed within the specific
    including the minimum value.

(2) On-chip power supply step-down circuit

The following products incorporate an internal power supply step-down circuit, whi automatically drops down the internal power supply voltage to the optimum interna level: the F-ZTAT A-mask versions of the H8S/2144 (HD64F2144A) and the mask versions of the H8S/2144, and H8S/2143 (HD6432144S and HD6432143S).

The voltage-stabilization capacitor (0.47 µF one or two connected in series) must be between the VCL (internal power supply step-down) and VSS pins.

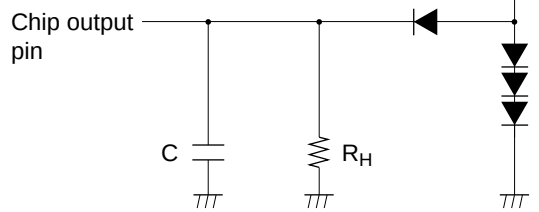
Figure 26.3 shows the connection of the external capacitors.

For the 5- or 4-V version, do not connect the VCL pin in a product incorporating an step-down circuit to the V_{CC} power supply. (Connect the VCC1 pin to the V_{CC} power usual.)

For the 3-V version, connect both the VCL and VCC1 pins to the system power sup

When changing from the F-ZTAT versions not incorporating an internal step-down the mask ROM versions incorporating an internal step-down circuit, the VCL pin ha pin location as the VCC2 pin.

Therefore, note that the circuit patterns differ between these two types of products.



C = 30 pF: All output
R_L = 2.4 kΩ
R_H = 12 kΩ

I/O timing test levels
- Low level: 0.8 V
- High level: 2.0 V

Figure 26.4 Output Load Circuit

26.7.2 Clock Timing

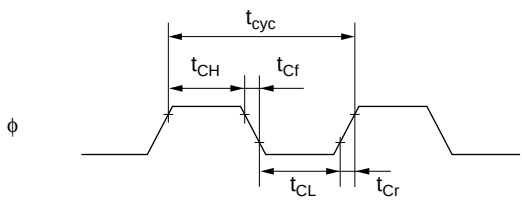


Figure 26.5 System Clock Timing

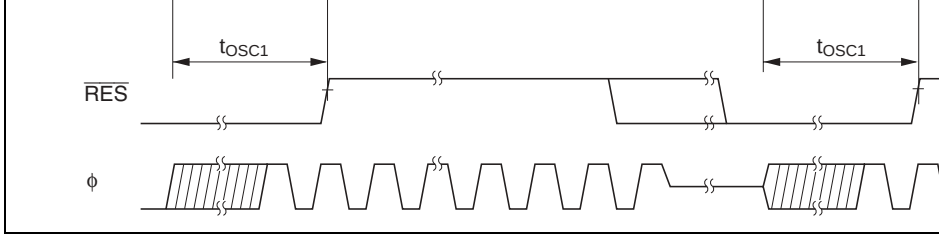


Figure 26.6 Oscillation Settling Timing

26.7.3 Control Signal Timing

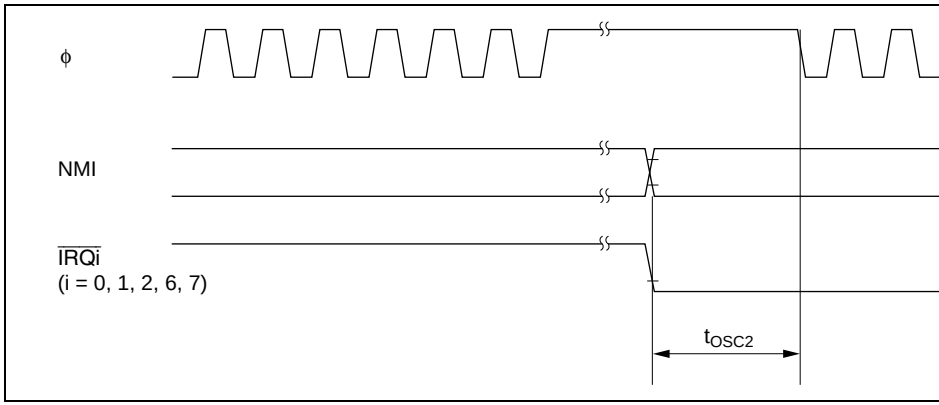


Figure 26.7 Oscillation Setting Timing (Exiting Software Standby Mod

Figure 26.8 Reset Input Timing

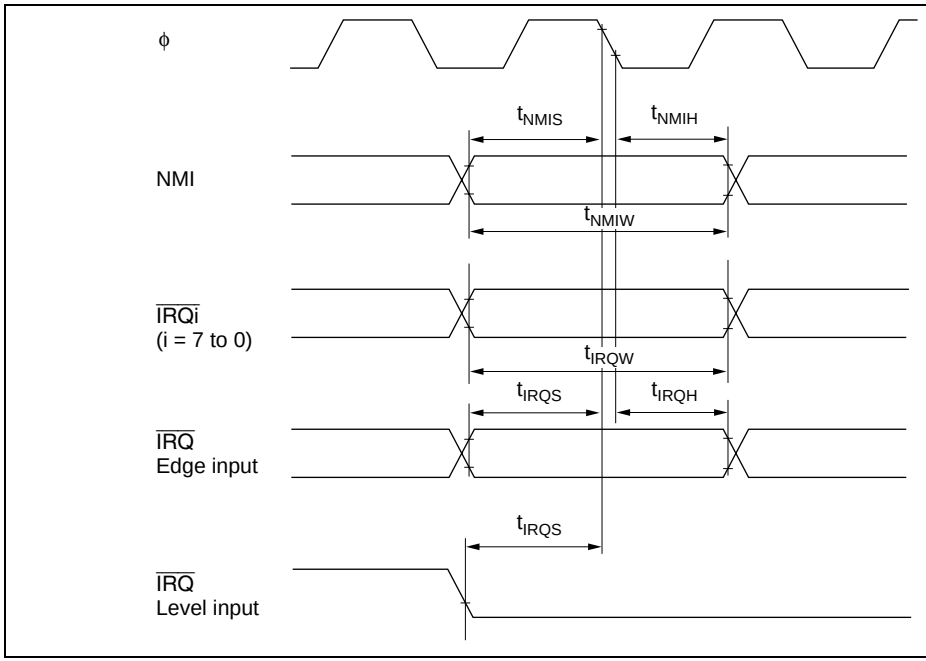
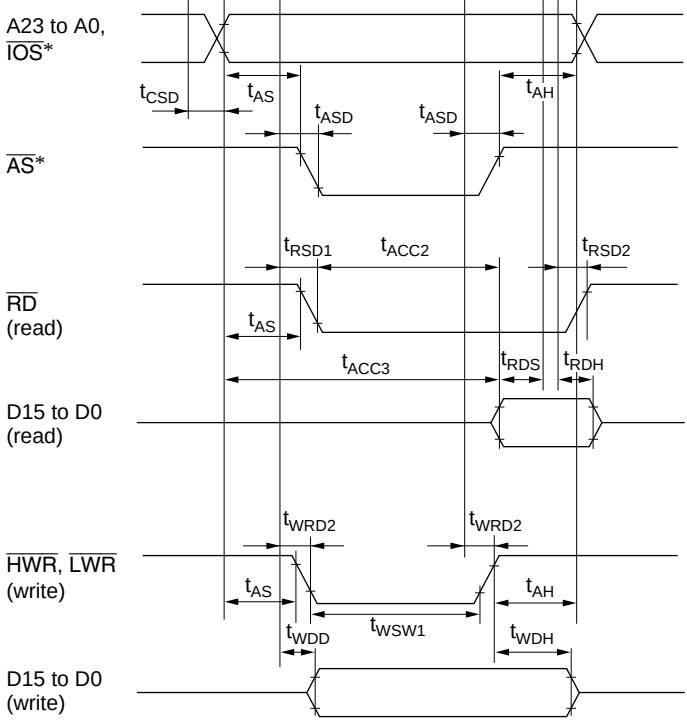
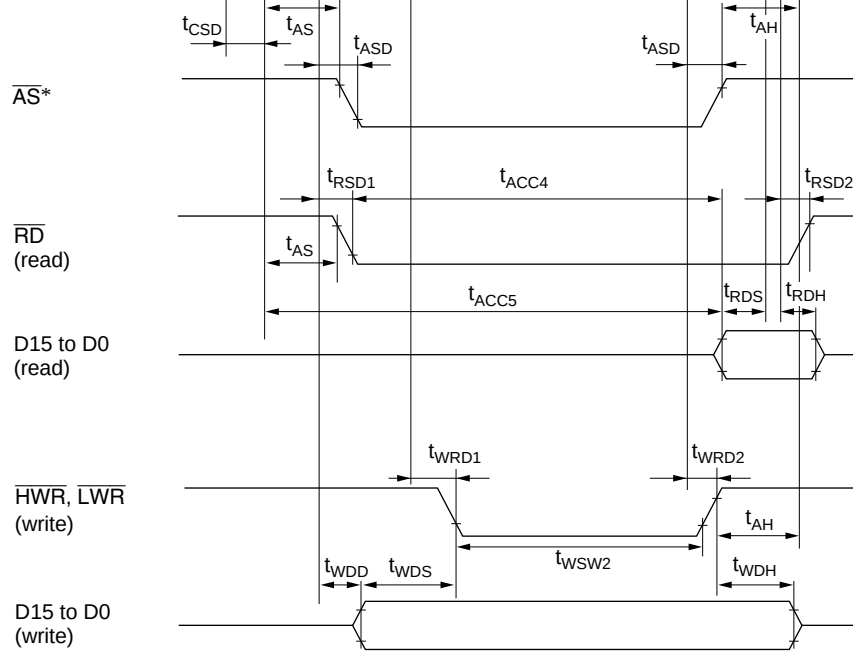


Figure 26.9 Interrupt Input Timing



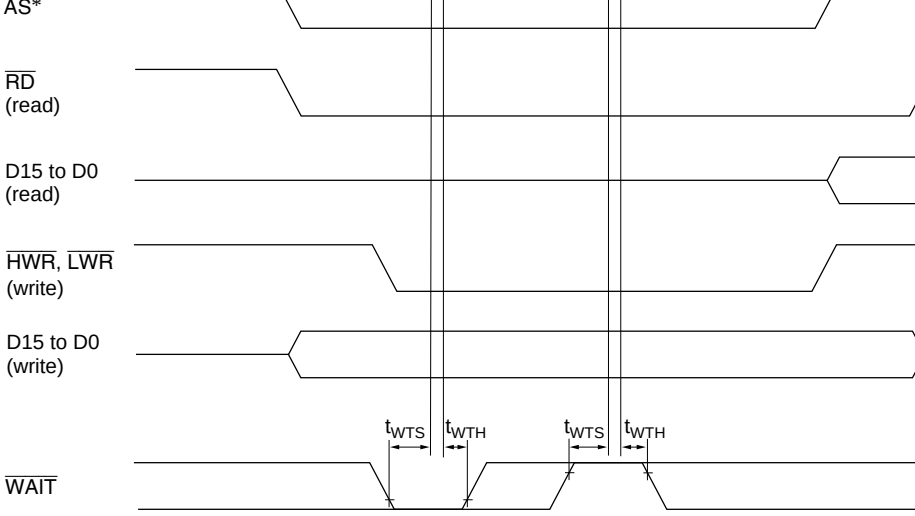
Note: * AS and IOS are the same pin. The function is selected by the IOSE bit in SYSC

Figure 26.10 Basic Bus Timing (Two-State Access)



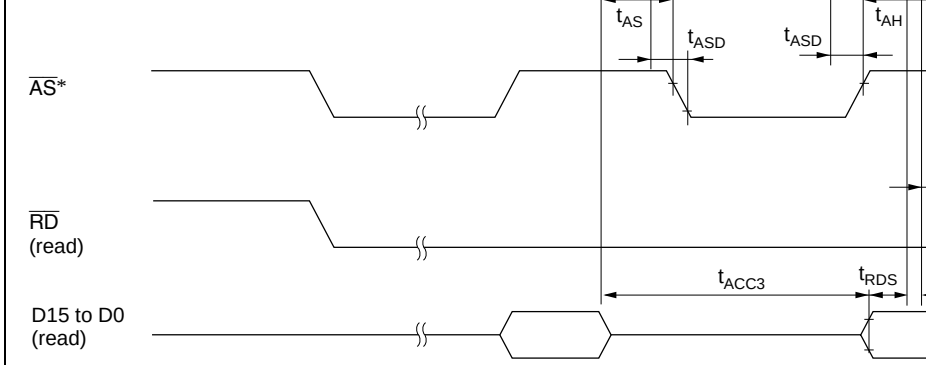
Note: * $\overline{AS}$ and $\overline{IOS}$ are the same pin. The function is selected by the IOSE bit in SYSCR

Figure 26.11 Basic Bus Timing (Three-State Access)



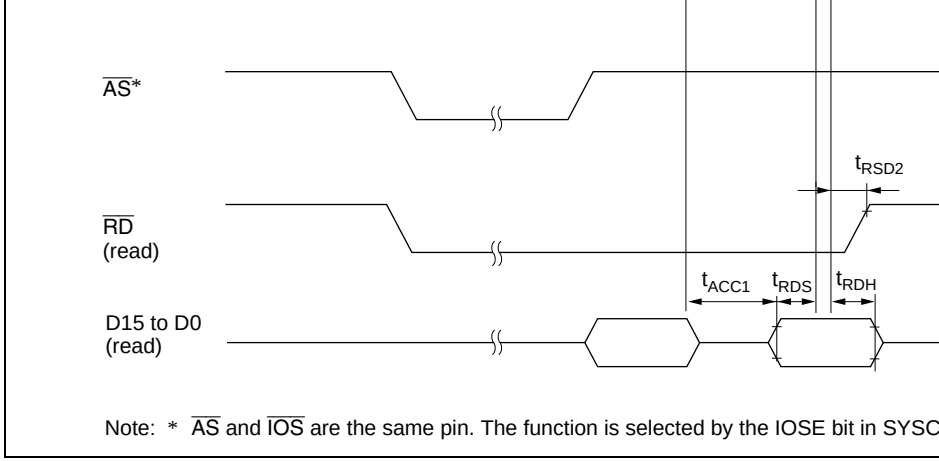
Note: * AS and IOS are the same pin. The function is selected by the IOSE bit in SYSCR.

Figure 26.12 Basic Bus Timing (Three-State Access with One Wait Stat



Note: * AS and IOS are the same pin. The function is selected by the IOSE bit in SYSCR.

Figure 26.13 Burst ROM Access Timing (Two-State Access)



Note: * AS and IOS are the same pin. The function is selected by the IOSE bit in SYSC

Figure 26.14 Burst ROM Access Timing (One-State Access)

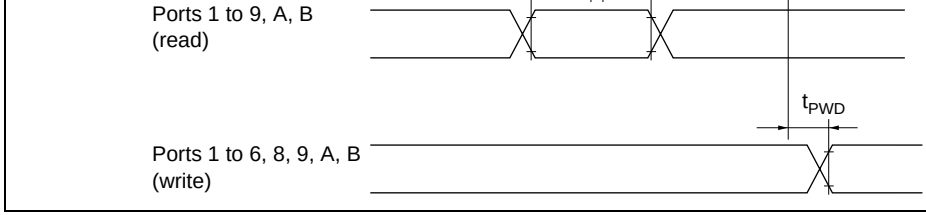


Figure 26.15 I/O Port Input/Output Timing

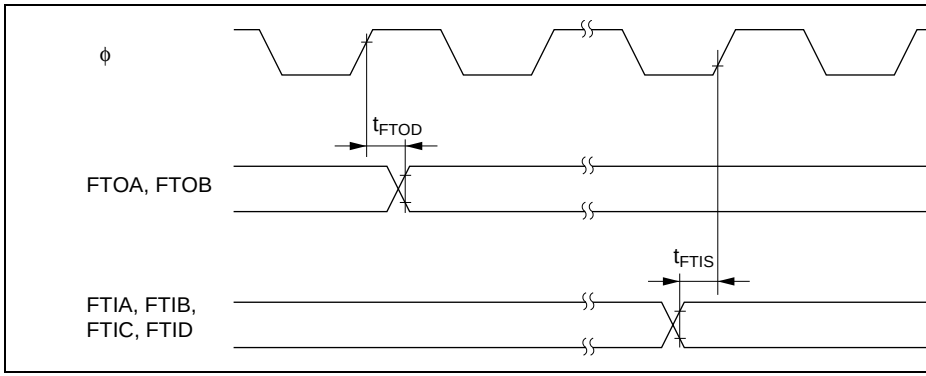


Figure 26.16 FRT Input/Output Timing

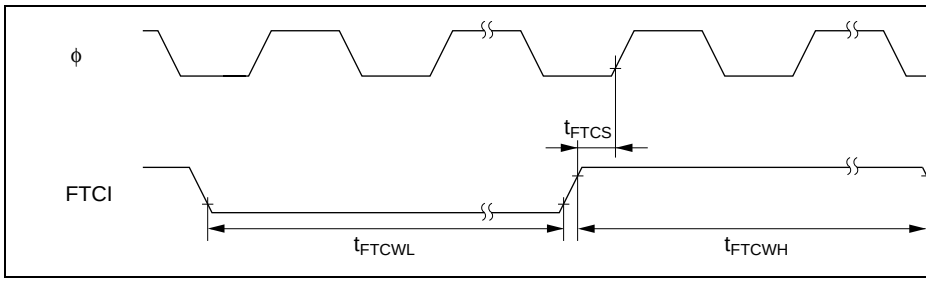


Figure 26.17 FRT Clock Input Timing

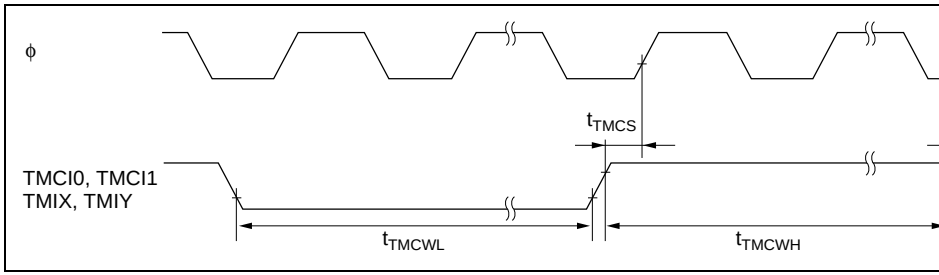


Figure 26.19 8-Bit Timer Clock Input Timing

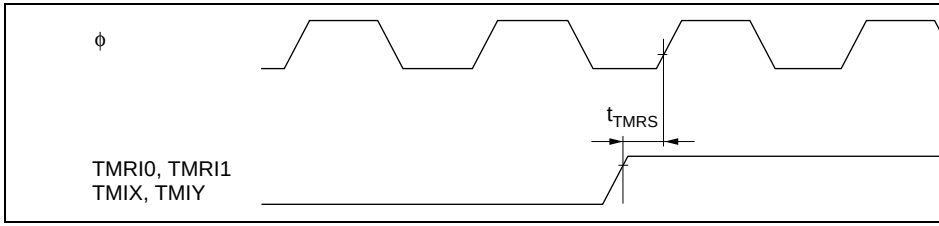


Figure 26.20 8-Bit Timer Reset Input Timing

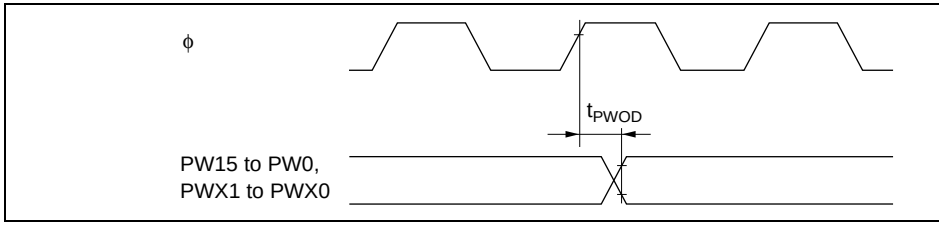


Figure 26.21 PWM, PWMX Output Timing

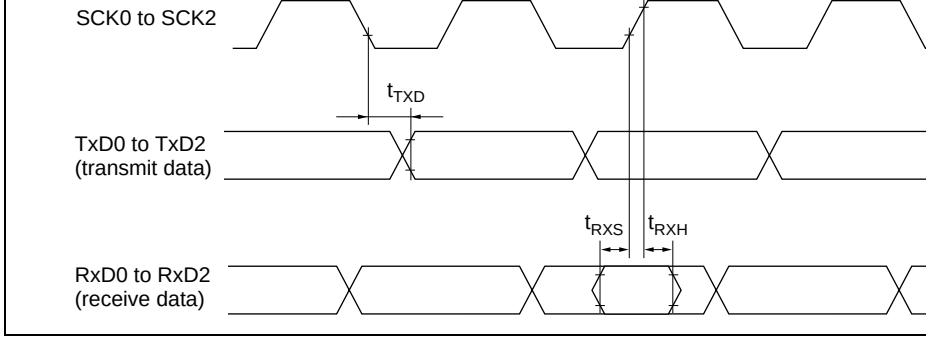


Figure 26.23 SCI Input/Output Timing (Synchronous Mode)

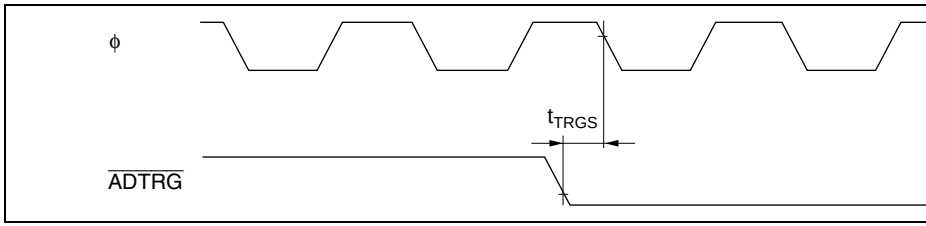


Figure 26.24 A/D Converter External Trigger Input Timing

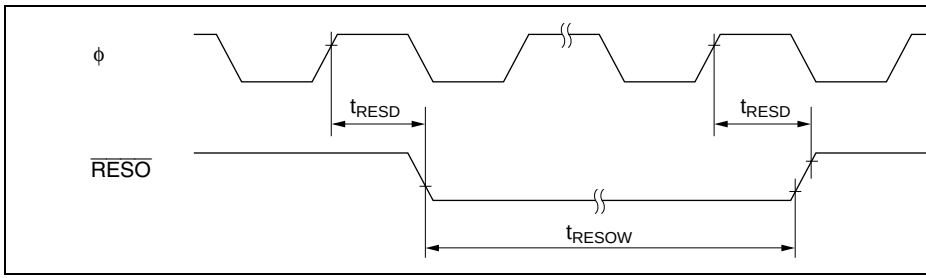
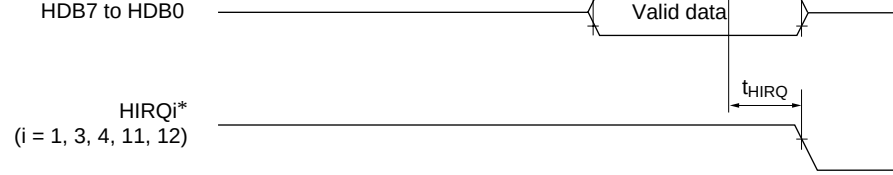


Figure 26.25 WDT Output Timing ($\overline{\text{RESO}}$)



Note: * The rising edge timing is the same as the port 4 and port B output tim
See figure 26.15.

Host interface write timing

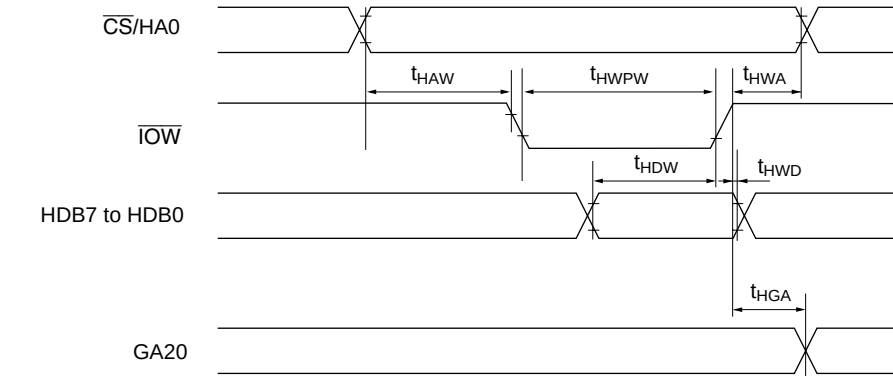
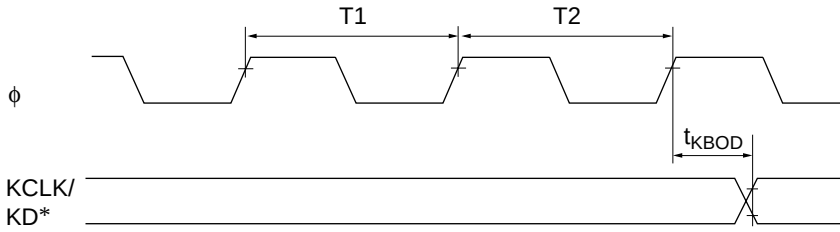
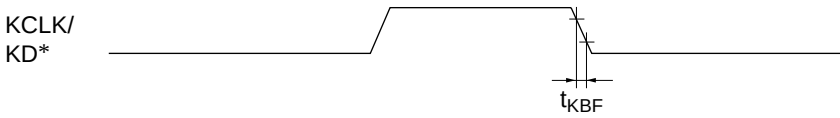


Figure 26.26 Host Interface Timing

2. Transmission (a)



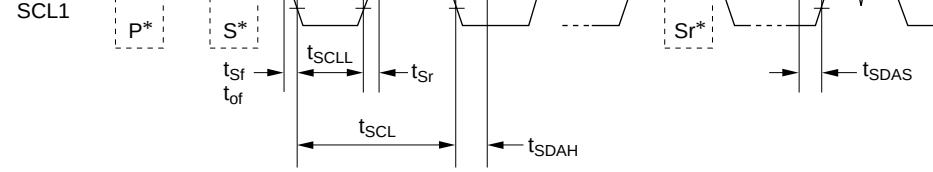
Transmission (b)



Note: ϕ shown here is the clock scaled by 1/N when the operating mode is active medium-speed mode.

* KCLK: PS2AC to PS2CC
KD: PS2AD to PS2CD

Figure 26.27 Keyboard Buffer Controller Timing



Note: * S, P, and Sr indicate the following conditions.

S: Start condition
P: Stop condition
Sr: Retransmission start condition

Figure 26.28 I²C Bus Interface Input/Output Timing (Option)

Rn	General register*1
ERn	General register (32-bit register)
MAC	Multiply-and-accumulate register (32-bit register)*2
(EAd)	Destination operand
(EAs)	Source operand
EXR	Extend 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
⊕	Exclusive logical OR
→	Transfer from left-hand operand to right-hand operand, or transition hand state to right-hand state
¬	NOT (logical complement)
( ) < >	Operand contents
:8/:16/:24/:32	8-, 16-, 24-, or 32-bit length

Notes: 1. General registers include 8-bit registers (R0H to R7H, R0L to R7L), 16-bit r to R7, E0 to E7), and 32-bit registers (ER0 to ER7).
2. MAC instructions cannot be used in this LSI.

MOV	MOV.B #xx:8,Rd	B	2								#xx:8→Rd8	—	—	↕	
	MOV.B Rs,Rd	B		2							Rs8→Rd8	—	—	↕	
	MOV.B @ERs,Rd	B			2						@ERs→Rd8	—	—	↕	
	MOV.B @(d:16,ERs),Rd	B				4					@(d:16,ERs)→Rd8	—	—	↕	
	MOV.B @(d:32,ERs),Rd	B				8					@(d:32,ERs)→Rd8	—	—	↕	
	MOV.B @ERs+,Rd	B					2				@ERs→Rd8,ERs32+1→ERs32	—	—	↕	
	MOV.B @aa:8,Rd	B						2			@aa:8→Rd8	—	—	↕	
	MOV.B @aa:16,Rd	B						4			@aa:16→Rd8	—	—	↕	
	MOV.B @aa:32,Rd	B						6			@aa:32→Rd8	—	—	↕	
	MOV.B Rs,@ERd	B			2						Rs8→@ERd	—	—	↕	
	MOV.B Rs,@(d:16,ERd)	B				4					Rs8→@(d:16,ERd)	—	—	↕	
	MOV.B Rs,@(d:32,ERd)	B				8					Rs8→@(d:32,ERd)	—	—	↕	
	MOV.B Rs,@-ERd	B					2				ERd32-1→ERd32,Rs8→@ERd	—	—	↕	
	MOV.B Rs,@aa:8	B						2			Rs8→@aa:8	—	—	↕	
	MOV.B Rs,@aa:16	B						4			Rs8→@aa:16	—	—	↕	
	MOV.B Rs,@aa:32	B						6			Rs8→@aa:32	—	—	↕	
	MOV.W #xx:16,Rd	W	4								#xx:16→Rd16	—	—	↕	
	MOV.W Rs,Rd	W		2							Rs16→Rd16	—	—	↕	
	MOV.W @ERs,Rd	W			2						@ERs→Rd16	—	—	↕	
	MOV.W @(d:16,ERs),Rd	W				4					@(d:16,ERs)→Rd16	—	—	↕	
	MOV.W @(d:32,ERs),Rd	W				8					@(d:32,ERs)→Rd16	—	—	↕	
	MOV.W @ERs+,Rd	W					2				@ERs→Rd16,ERs32+2→ERs32	—	—	↕	
	MOV.W @aa:16,Rd	W						4			@aa:16→Rd16	—	—	↕	
	MOV.W @aa:32,Rd	W						6			@aa:32→Rd16	—	—	↕	
	MOV.W Rs,@ERd	W			2						Rs16→@ERd	—	—	↕	
	MOV.W Rs,@(d:16,ERd)	W				4					Rs16→@(d:16,ERd)	—	—	↕	
	MOV.W Rs,@(d:32,ERd)	W				8					Rs16→@(d:32,ERd)	—	—	↕	
	MOV.W Rs,@-ERd	W					2				ERd32-2→ERd32,Rs16→@ERd	—	—	↕	
	MOV.W Rs,@aa:16	W						4			Rs16→@aa:16	—	—	↕	
	MOV.W Rs,@aa:32	W						6			Rs16→@aa:32	—	—	↕	

	MOV.L @(d:16,ERs),ERd	L				6						@(d:16,ERs)→ERd32	—	—	↕	↕
	MOV.L @(d:32,ERs),ERd	L				10						@(d:32,ERs)→ERd32	—	—	↕	↕
	MOV.L @ERs+,ERd	L					4					@ERs→ERd32,ERs32+4→ERs32	—	—	↕	↕
	MOV.L @aa:16,ERd	L						6				@aa:16→ERd32	—	—	↕	↕
	MOV.L @aa:32,ERd	L						8				@aa:32→ERd32	—	—	↕	↕
	MOV.L ERs,@ERd	L			4							ERs32→@ERd	—	—	↕	↕
	MOV.L ERs,@(d:16,ERd)	L				6						ERs32→@(d:16,ERd)	—	—	↕	↕
	MOV.L ERs,@(d:32,ERd)	L				10						ERs32→@(d:32,ERd)	—	—	↕	↕
	MOV.L ERs,@-ERd	L					4					ERd32-4→ERd32,ERs32→@ERd	—	—	↕	↕
	MOV.L ERs,@aa:16	L						6				ERs32→@aa:16	—	—	↕	↕
	MOV.L ERs,@aa:32	L						8				ERs32→@aa:32	—	—	↕	↕
POP	POP.W Rn	W									2	@SP→Rn16,SP+2→SP	—	—	↕	↕
	POP.L ERn	L									4	@SP→ERn32,SP+4→SP	—	—	↕	↕
PUSH	PUSH.W Rn	W									2	SP-2→SP,Rn16→@SP	—	—	↕	↕
	PUSH.L ERn	L									4	SP-4→SP,ERn32→@SP	—	—	↕	↕
LDM*4	LDM @SP+,(ERm-ERn)	L									4	(@SP→ERn32,SP+4→SP) Repeated for each restored register.	—	—	—	—
STM*4	STM (ERm-ERn),@-SP	L									4	(SP-4→SP,ERn32→@SP) Repeated for each saved register.	—	—	—	—
MOVFPE	MOVFPE @aa:16,Rd	Cannot be used with this LSI.														
MOVTPE	MOVTPE Rs,@aa:16															

	ADD.B Rs,Rd	B		2							Rd8+Rs8→Rd8	—	↕	↕		
	ADD.W #xx:16,Rd	W	4								Rd16+#xx:16→Rd16	—	[3]	↕		
	ADD.W Rs,Rd	W		2							Rd16+Rs16→Rd16	—	[3]	↕		
	ADD.L #xx:32,ERd	L	6								ERd32+#xx:32→ERd32	—	[4]	↕		
	ADD.L ERs,ERd	L		2							ERd32+ERs32→ERd32	—	[4]	↕		
ADDX	ADDX #xx:8,Rd	B	2								Rd8+#xx:8+C→Rd8	—	↕	↕		[5]
	ADDX Rs,Rd	B		2							Rd8+Rs8+C→Rd8	—	↕	↕		[5]
ADDS	ADDS #1,ERd	L		2							ERd32+1→ERd32	—	—	—		
	ADDS #2,ERd	L		2							ERd32+2→ERd32	—	—	—		
	ADDS #4,ERd	L		2							ERd32+4→ERd32	—	—	—		
INC	INC.B Rd	B		2							Rd8+1→Rd8	—	—	↕		
	INC.W #1,Rd	W		2							Rd16+1→Rd16	—	—	↕		
	INC.W #2,Rd	W		2							Rd16+2→Rd16	—	—	↕		
	INC.L #1,ERd	L		2							ERd32+1→ERd32	—	—	↕		
	INC.L #2,ERd	L		2							ERd32+2→ERd32	—	—	↕		
DAA	DAA Rd	B		2							Rd8 decimal adjust →Rd8	—	*	↕		
SUB	SUB.B Rs,Rd	B		2							Rd8-Rs8→Rd8	—	↕	↕		
	SUB.W #xx:16,Rd	W	4								Rd16-#xx:16→Rd16	—	[3]	↕		
	SUB.W Rs,Rd	W		2							Rd16-Rs16→Rd16	—	[3]	↕		
	SUB.L #xx:32,ERd	L	6								ERd32-#xx:32→ERd32	—	[4]	↕		
	SUB.L ERs,ERd	L		2							ERd32-ERs32→ERd32	—	[4]	↕		
SUBX	SUBX #xx:8,Rd	B	2								Rd8-#xx:8-C→Rd8	—	↕	↕		[5]
	SUBX Rs,Rd	B		2							Rd8-Rs8-C→Rd8	—	↕	↕		[5]
SUBS	SUBS #1,ERd	L		2							ERd32-1→ERd32	—	—	—		
	SUBS #2,ERd	L		2							ERd32-2→ERd32	—	—	—		
	SUBS #4,ERd	L		2							ERd32-4→ERd32	—	—	—		
DEC	DEC.B Rd	B		2							Rd8-1→Rd8	—	—	↕		
	DEC.W #1,Rd	W		2							Rd16-1→Rd16	—	—	↕		
	DEC.W #2,Rd	W		2							Rd16-2→Rd16	—	—	↕		
	DEC.L #1,ERd	L		2							ERd32-1→ERd32	—	—	↕		
	DEC.L #2,ERd	L		2							ERd32-2→ERd32	—	—	↕		

	MULXU.W Rs,ERd	W		2								Rd16×Rs16→ERd32 (unsigned multiplication)	—	—	—	—
MULXS	MULXS.B Rs,Rd	B		4								Rd8×Rs8→Rd16 (signed multiplication)	—	—	↕	↕
	MULXS.W Rs,ERd	W		4								Rd16×Rs16→ERd32 (signed multiplication)	—	—	↕	↕
DIVXU	DIVXU.B Rs,Rd	B		2								Rd16÷Rs8→Rd16 (RdH: remainder, RdL: quotient) (unsigned division)	—	—	[6]	[7]
	DIVXU.W Rs,ERd	W		2								ERd32÷Rs16→ERd32 (Ed: remainder, Rd: quotient) (unsigned division)	—	—	[6]	[7]
DIVXS	DIVXS.B Rs,Rd	B		4								Rd16÷Rs8→Rd16 (RdH: remainder, RdL: quotient) (signed division)	—	—	[8]	[7]
	DIVXS.W Rs,ERd	W		4								ERd32÷Rs16→ERd32 (Ed: remainder, Rd: quotient) (signed division)	—	—	[8]	[7]
CMP	CMP.B #xx:8,Rd	B	2									Rd8-#xx:8	—	↕	↕	↕
	CMP.B Rs,Rd	B		2								Rd8-Rs8	—	↕	↕	↕
	CMP.W #xx:16,Rd	W	4									Rd16-#xx:16	—	[3]	↕	↕
	CMP.W Rs,Rd	W		2								Rd16-Rs16	—	[3]	↕	↕
	CMP.L #xx:32,ERd	L	6									ERd32-#xx:32	—	[4]	↕	↕
	CMP.L ERs,ERd	L		2								ERd32-ERs32	—	[4]	↕	↕
NEG	NEG.B Rd	B		2								0-Rd8→Rd8	—	↕	↕	↕
	NEG.W Rd	W		2								0-Rd16→Rd16	—	↕	↕	↕
	NEG.L ERd	L		2								0-ERd32→ERd32	—	↕	↕	↕
EXTU	EXTU.W Rd	W		2								0 → (<bits 15 to 8> of Rd16)	—	—	0	↕
	EXTU.L ERd	L		2								0 → (<bits 31 to 16> of ERd32)	—	—	0	↕
EXTS	EXTS.W Rd	W		2								(<bit 7> of Rd16) → (<bits 15 to 8> of Rd16)	—	—	↕	↕
	EXTS.L ERd	L		2								(<bit 15> of ERd32) → (<bits 31 to 16> of ERd32)	—	—	↕	↕
TAS	TAS @ERd*3	B			4							@ERd-0 → CCR set, (1) → (<bit 7> of @ERd)	—	—	↕	↕

CLRMAC	CLRMAC	
LDMAC	LDMAC ERs,MACH	
	LDMAC ERs,MACL	
STMAC	STMAC MACH,ERd	
	STMAC MACL,ERd	

	AND.B Rs,Rd	B		2							Rd8∧Rs8→Rd8	—	—	↕	↕
	AND.W #xx:16,Rd	W	4								Rd16∧#xx:16→Rd16	—	—	↕	↕
	AND.W Rs,Rd	W		2							Rd16∧Rs16→Rd16	—	—	↕	↕
	AND.L #xx:32,ERd	L	6								ERd32∧#xx:32→ERd32	—	—	↕	↕
	AND.L ERs,ERd	L		4							ERd32∧ERs32→ERd32	—	—	↕	↕
OR	OR.B #xx:8,Rd	B	2								Rd8∨#xx:8→Rd8	—	—	↕	↕
	OR.B Rs,Rd	B		2							Rd8∨Rs8→Rd8	—	—	↕	↕
	OR.W #xx:16,Rd	W	4								Rd16∨#xx:16→Rd16	—	—	↕	↕
	OR.W Rs,Rd	W		2							Rd16∨Rs16→Rd16	—	—	↕	↕
	OR.L #xx:32,ERd	L	6								ERd32∨#xx:32→ERd32	—	—	↕	↕
	OR.L ERs,ERd	L		4							ERd32∨ERs32→ERd32	—	—	↕	↕
XOR	XOR.B #xx:8,Rd	B	2								Rd8⊕#xx:8→Rd8	—	—	↕	↕
	XOR.B Rs,Rd	B		2							Rd8⊕Rs8→Rd8	—	—	↕	↕
	XOR.W #xx:16,Rd	W	4								Rd16⊕#xx:16→Rd16	—	—	↕	↕
	XOR.W Rs,Rd	W		2							Rd16⊕Rs16→Rd16	—	—	↕	↕
	XOR.L #xx:32,ERd	L	6								ERd32⊕#xx:32→ERd32	—	—	↕	↕
	XOR.L ERs,ERd	L		4							ERd32⊕ERs32→ERd32	—	—	↕	↕
NOT	NOT.B Rd	B		2							¬ Rd8→Rd8	—	—	↕	↕
	NOT.W Rd	W		2							¬ Rd16→Rd16	—	—	↕	↕
	NOT.L ERd	L		2							¬ ERd32→ERd32	—	—	↕	↕

	SHAL.B #2,Rd	B		2										—	—	↕
	SHAL.W Rd	W		2										—	—	↕
	SHAL.W #2,Rd	W		2										—	—	↕
	SHAL.L ERd	L		2										—	—	↕
	SHAL.L #2,ERd	L		2										—	—	↕
SHAR	SHAR.B Rd	B		2										—	—	↕
	SHAR.B #2,Rd	B		2										—	—	↕
	SHAR.W Rd	W		2										—	—	↕
	SHAR.W #2,Rd	W		2										—	—	↕
	SHAR.L ERd	L		2										—	—	↕
	SHAR.L #2,ERd	L		2										—	—	↕
SHLL	SHLL.B Rd	B		2										—	—	↕
	SHLL.B #2,Rd	B		2										—	—	↕
	SHLL.W Rd	W		2										—	—	↕
	SHLL.W #2,Rd	W		2										—	—	↕
	SHLL.L ERd	L		2										—	—	↕
	SHLL.L #2,ERd	L		2										—	—	↕
SHLR	SHLR.B Rd	B		2										—	—	0
	SHLR.B #2,Rd	B		2										—	—	0
	SHLR.W Rd	W		2										—	—	0
	SHLR.W #2,Rd	W		2										—	—	0
	SHLR.L ERd	L		2										—	—	0
	SHLR.L #2,ERd	L		2										—	—	0
ROTXL	ROTXL.B Rd	B		2										—	—	↕
	ROTXL.B #2,Rd	B		2										—	—	↕
	ROTXL.W Rd	W		2										—	—	↕
	ROTXL.W #2,Rd	W		2										—	—	↕
	ROTXL.L ERd	L		2										—	—	↕
	ROTXL.L #2,ERd	L		2										—	—	↕

	ROTXR.W #2,Rd	W		2															
	ROTXR.L ERd	L		2															
	ROTXR.L #2,ERd	L		2															
ROTL	ROTL.B Rd	B		2															
	ROTL.B #2,Rd	B		2															
	ROTL.W Rd	W		2															
	ROTL.W #2,Rd	W		2															
	ROTL.L ERd	L		2															
	ROTL.L #2,ERd	L		2															
ROTR	ROTR.B Rd	B		2															
	ROTR.B #2,Rd	B		2															
	ROTR.W Rd	W		2															
	ROTR.W #2,Rd	W		2															
	ROTR.L ERd	L		2															
	ROTR.L #2,ERd	L		2															

	BSET #xx:3,@ERd	B			4						(#xx:3 of @ERd)←1	—	—	—	-
	BSET #xx:3,@aa:8	B						4			(#xx:3 of @aa:8)←1	—	—	—	-
	BSET #xx:3,@aa:16	B						6			(#xx:3 of @aa:16)←1	—	—	—	-
	BSET #xx:3,@aa:32	B						8			(#xx:3 of @aa:32)←1	—	—	—	-
	BSET Rn,Rd	B		2							(Rn8 of Rd8)←1	—	—	—	-
	BSET Rn,@ERd	B			4						(Rn8 of @ERd)←1	—	—	—	-
	BSET Rn,@aa:8	B						4			(Rn8 of @aa:8)←1	—	—	—	-
	BSET Rn,@aa:16	B						6			(Rn8 of @aa:16)←1	—	—	—	-
	BSET Rn,@aa:32	B						8			(Rn8 of @aa:32)←1	—	—	—	-
BCLR	BCLR #xx:3,Rd	B		2							(#xx:3 of Rd8)←0	—	—	—	-
	BCLR #xx:3,@ERd	B			4						(#xx:3 of @ERd)←0	—	—	—	-
	BCLR #xx:3,@aa:8	B						4			(#xx:3 of @aa:8)←0	—	—	—	-
	BCLR #xx:3,@aa:16	B						6			(#xx:3 of @aa:16)←0	—	—	—	-
	BCLR #xx:3,@aa:32	B						8			(#xx:3 of @aa:32)←0	—	—	—	-
	BCLR Rn,Rd	B		2							(Rn8 of Rd8)←0	—	—	—	-
	BCLR Rn,@ERd	B			4						(Rn8 of @ERd)←0	—	—	—	-
	BCLR Rn,@aa:8	B						4			(Rn8 of @aa:8)←0	—	—	—	-
	BCLR Rn,@aa:16	B						6			(Rn8 of @aa:16)←0	—	—	—	-
	BCLR Rn,@aa:32	B						8			(Rn8 of @aa:32)←0	—	—	—	-
BNOT	BNOT #xx:3,Rd	B		2							(#xx:3 of Rd8)← [¬ (#xx:3 of Rd8)]	—	—	—	-
	BNOT #xx:3,@ERd	B			4						(#xx:3 of @ERd)← [¬ (#xx:3 of @ERd)]	—	—	—	-
	BNOT #xx:3,@aa:8	B						4			(#xx:3 of @aa:8)← [¬ (#xx:3 of @aa:8)]	—	—	—	-
	BNOT #xx:3,@aa:16	B						6			(#xx:3 of @aa:16)← [¬ (#xx:3 of @aa:16)]	—	—	—	-
	BNOT #xx:3,@aa:32	B						8			(#xx:3 of @aa:32)← [¬ (#xx:3 of @aa:32)]	—	—	—	-
	BNOT Rn,Rd	B		2							(Rn8 of Rd8)← [¬ (Rn8 of Rd8)]	—	—	—	-
	BNOT Rn,@ERd	B			4						(Rn8 of @ERd)← [¬ (Rn8 of @ERd)]	—	—	—	-
	BNOT Rn,@aa:8	B						4			(Rn8 of @aa:8)← [¬ (Rn8 of @aa:8)]	—	—	—	-
	BNOT Rn,@aa:16	B						6			(Rn8 of @aa:16)← [¬ (Rn8 of @aa:16)]	—	—	—	-
	BNOT Rn,@aa:32	B						8			(Rn8 of @aa:32)← [¬ (Rn8 of @aa:32)]	—	—	—	-

	BTST #xx:3,@aa:16	B					6				¬ (#xx:3 of @aa:16)→Z	—	—	—	↕
	BTST #xx:3,@aa:32	B					8				¬ (#xx:3 of @aa:32)→Z	—	—	—	↕
	BTST Rn,Rd	B		2							¬ (Rn8 of Rd8)→Z	—	—	—	↕
	BTST Rn,@ERd	B			4						¬ (Rn8 of @ERd)→Z	—	—	—	↕
	BTST Rn,@aa:8	B					4				¬ (Rn8 of @aa:8)→Z	—	—	—	↕
	BTST Rn,@aa:16	B					6				¬ (Rn8 of @aa:16)→Z	—	—	—	↕
	BTST Rn,@aa:32	B					8				¬ (Rn8 of @aa:32)→Z	—	—	—	↕
BLD	BLD #xx:3,Rd	B		2							(#xx:3 of Rd8)→C	—	—	—	—
	BLD #xx:3,@ERd	B			4						(#xx:3 of @ERd)→C	—	—	—	—
	BLD #xx:3,@aa:8	B					4				(#xx:3 of @aa:8)→C	—	—	—	—
	BLD #xx:3,@aa:16	B					6				(#xx:3 of @aa:16)→C	—	—	—	—
	BLD #xx:3,@aa:32	B					8				(#xx:3 of @aa:32)→C	—	—	—	—
BILD	BILD #xx:3,Rd	B		2							¬ (#xx:3 of Rd8)→C	—	—	—	—
	BILD #xx:3,@ERd	B			4						¬ (#xx:3 of @ERd)→C	—	—	—	—
	BILD #xx:3,@aa:8	B					4				¬ (#xx:3 of @aa:8)→C	—	—	—	—
	BILD #xx:3,@aa:16	B					6				¬ (#xx:3 of @aa:16)→C	—	—	—	—
	BILD #xx:3,@aa:32	B					8				¬ (#xx:3 of @aa:32)→C	—	—	—	—
BST	BST #xx:3,Rd	B		2							C→(#xx:3 of Rd8)	—	—	—	—
	BST #xx:3,@ERd	B			4						C→(#xx:3 of @ERd)	—	—	—	—
	BST #xx:3,@aa:8	B					4				C→(#xx:3 of @aa:8)	—	—	—	—
	BST #xx:3,@aa:16	B					6				C→(#xx:3 of @aa:16)	—	—	—	—
	BST #xx:3,@aa:32	B					8				C→(#xx:3 of @aa:32)	—	—	—	—
BIST	BIST #xx:3,Rd	B		2							¬ C→(#xx:3 of Rd8)	—	—	—	—
	BIST #xx:3,@ERd	B			4						¬ C→(#xx:3 of @ERd)	—	—	—	—
	BIST #xx:3,@aa:8	B					4				¬ C→(#xx:3 of @aa:8)	—	—	—	—
	BIST #xx:3,@aa:16	B					6				¬ C→(#xx:3 of @aa:16)	—	—	—	—
	BIST #xx:3,@aa:32	B					8				¬ C→(#xx:3 of @aa:32)	—	—	—	—

	BAND #xx:3,@aa:16	B						6				C∧ (#xx:3 of @aa:16)→C	—	—	—
	BAND #xx:3,@aa:32	B						8				C∧ (#xx:3 of @aa:32)→C	—	—	—
BIAND	BIAND #xx:3,Rd	B		2								C∧ [¬ (#xx:3 of Rd8)]→C	—	—	—
	BIAND #xx:3,@ERd	B			4							C∧ [¬ (#xx:3 of @ERd)]→C	—	—	—
	BIAND #xx:3,@aa:8	B						4				C∧ [¬ (#xx:3 of @aa:8)]→C	—	—	—
	BIAND #xx:3,@aa:16	B						6				C∧ [¬ (#xx:3 of @aa:16)]→C	—	—	—
	BIAND #xx:3,@aa:32	B						8				C∧ [¬ (#xx:3 of @aa:32)]→C	—	—	—
BOR	BOR #xx:3,Rd	B		2								C∨ (#xx:3 of Rd8)→C	—	—	—
	BOR #xx:3,@ERd	B			4							C∨ (#xx:3 of @ERd)→C	—	—	—
	BOR #xx:3,@aa:8	B						4				C∨ (#xx:3 of @aa:8)→C	—	—	—
	BOR #xx:3,@aa:16	B						6				C∨ (#xx:3 of @aa:16)→C	—	—	—
	BOR #xx:3,@aa:32	B						8				C∨ (#xx:3 of @aa:32)→C	—	—	—
BIOR	BIOR #xx:3,Rd	B		2								C∨ [¬ (#xx:3 of Rd8)]→C	—	—	—
	BIOR #xx:3,@ERd	B			4							C∨ [¬ (#xx:3 of @ERd)]→C	—	—	—
	BIOR #xx:3,@aa:8	B						4				C∨ [¬ (#xx:3 of @aa:8)]→C	—	—	—
	BIOR #xx:3,@aa:16	B						6				C∨ [¬ (#xx:3 of @aa:16)]→C	—	—	—
	BIOR #xx:3,@aa:32	B						8				C∨ [¬ (#xx:3 of @aa:32)]→C	—	—	—
BXOR	BXOR #xx:3,Rd	B		2								C⊕ (#xx:3 of Rd8)→C	—	—	—
	BXOR #xx:3,@ERd	B			4							C⊕ (#xx:3 of @ERd)→C	—	—	—
	BXOR #xx:3,@aa:8	B						4				C⊕ (#xx:3 of @aa:8)→C	—	—	—
	BXOR #xx:3,@aa:16	B						6				C⊕ (#xx:3 of @aa:16)→C	—	—	—
	BXOR #xx:3,@aa:32	B						8				C⊕ (#xx:3 of @aa:32)→C	—	—	—
BIXOR	BIXOR #xx:3,Rd	B		2								C⊕ [¬ (#xx:3 of Rd8)]→C	—	—	—
	BIXOR #xx:3,@ERd	B			4							C⊕ [¬ (#xx:3 of @ERd)]→C	—	—	—
	BIXOR #xx:3,@aa:8	B						4				C⊕ [¬ (#xx:3 of @aa:8)]→C	—	—	—
	BIXOR #xx:3,@aa:16	B						6				C⊕ [¬ (#xx:3 of @aa:16)]→C	—	—	—
	BIXOR #xx:3,@aa:32	B						8				C⊕ [¬ (#xx:3 of @aa:32)]→C	—	—	—

	BRA d:16(BT d:16)	—						4			PC←PC+d else next;		—	—	—	—
	BRN d:8(BF d:8)	—						2				Never	—	—	—	—
	BRN d:16(BF d:16)	—						4					—	—	—	—
	BHI d:8	—						2				C∨Z=0	—	—	—	—
	BHI d:16	—						4					—	—	—	—
	BLS d:8	—						2				C∨Z=1	—	—	—	—
	BLS d:16	—						4					—	—	—	—
	BCC d:8(BHS d:8)	—						2				C=0	—	—	—	—
	BCC d:16(BHS d:16)	—						4					—	—	—	—
	BCS d:8(BLO d:8)	—						2				C=1	—	—	—	—
	BCS d:16(BLO d:16)	—						4					—	—	—	—
	BNE d:8	—						2				Z=0	—	—	—	—
	BNE d:16	—						4					—	—	—	—
	BEQ d:8	—						2				Z=1	—	—	—	—
	BEQ d:16	—						4					—	—	—	—
	BVC d:8	—						2				V=0	—	—	—	—
	BVC d:16	—						4					—	—	—	—
	BVS d:8	—						2				V=1	—	—	—	—
	BVS d:16	—						4					—	—	—	—
	BPL d:8	—						2				N=0	—	—	—	—
	BPL d:16	—						4					—	—	—	—
	BMI d:8	—						2				N=1	—	—	—	—
	BMI d:16	—						4					—	—	—	—
	BGE d:8	—						2				N⊕V=0	—	—	—	—
	BGE d:16	—						4					—	—	—	—
	BLT d:8	—						2				N⊕V=1	—	—	—	—
	BLT d:16	—						4					—	—	—	—
	BGT d:8	—						2				Z∨(N⊕V)=0	—	—	—	—
	BGT d:16	—						4					—	—	—	—
	BLE d:8	—						2				Z∨(N⊕V)=1	—	—	—	—
	BLE d:16	—						4					—	—	—	—

BSR	BSR d:8	—							2			PC→@-SP,PC←PC+d:8	—	—	—	–
	BSR d:16	—							4			PC→@-SP,PC←PC+d:16	—	—	—	–
JSR	JSR @ERn	—			2							PC→@-SP,PC←ERn	—	—	—	–
	JSR @aa:24	—						4				PC→@-SP,PC←aa:24	—	—	—	–
	JSR @@aa:8	—								2		PC→@-SP,PC←@aa:8	—	—	—	–
RTS	RTS	—									2	PC←@SP+	—	—	—	–

											EXR→@-SP,<vector>→PC				
RTE	RTE	—									EXR←@SP+,CCR←@SP+, PC←@SP+	↕	↕	↕	↕
SLEEP	SLEEP	—									Transition to power-down state	—	—	—	—
LDC	LDC #xx:8,CCR	B	2								#xx:8→CCR	↕	↕	↕	↕
	LDC #xx:8,EXR	B	4								#xx:8→EXR	—	—	—	—
	LDC Rs,CCR	B		2							Rs8→CCR	↕	↕	↕	↕
	LDC Rs,EXR	B		2							Rs8→EXR	—	—	—	—
	LDC @ERs,CCR	W			4						@ERs→CCR	↕	↕	↕	↕
	LDC @ERs,EXR	W			4						@ERs→EXR	—	—	—	—
	LDC @(d:16,ERs),CCR	W				6					@(d:16,ERs)→CCR	↕	↕	↕	↕
	LDC @(d:16,ERs),EXR	W				6					@(d:16,ERs)→EXR	—	—	—	—
	LDC @(d:32,ERs),CCR	W				10					@(d:32,ERs)→CCR	↕	↕	↕	↕
	LDC @(d:32,ERs),EXR	W				10					@(d:32,ERs)→EXR	—	—	—	—
	LDC @ERs+,CCR	W					4				@ERs→CCR,ERs32+2→ERs32	↕	↕	↕	↕
	LDC @ERs+,EXR	W					4				@ERs→EXR,ERs32+2→ERs32	—	—	—	—
	LDC @aa:16,CCR	W						6			@aa:16→CCR	↕	↕	↕	↕
	LDC @aa:16,EXR	W						6			@aa:16→EXR	—	—	—	—
	LDC @aa:32,CCR	W						8			@aa:32→CCR	↕	↕	↕	↕
	LDC @aa:32,EXR	W						8			@aa:32→EXR	—	—	—	—

	STC EXR,@ERd	W			4						EXR→@ERd	—	—	—	—	—
	STC CCR,@(d:16,ERd)	W				6					CCR→@(d:16,ERd)	—	—	—	—	—
	STC EXR,@(d:16,ERd)	W				6					EXR→@(d:16,ERd)	—	—	—	—	—
	STC CCR,@(d:32,ERd)	W				10					CCR→@(d:32,ERd)	—	—	—	—	—
	STC EXR,@(d:32,ERd)	W				10					EXR→@(d:32,ERd)	—	—	—	—	—
	STC CCR,@-ERd	W					4				ERd32-2→ERd32,CCR→@ERd	—	—	—	—	—
	STC EXR,@-ERd	W					4				ERd32-2→ERd32,EXR→@ERd	—	—	—	—	—
	STC CCR,@aa:16	W						6			CCR→@aa:16	—	—	—	—	—
	STC EXR,@aa:16	W						6			EXR→@aa:16	—	—	—	—	—
	STC CCR,@aa:32	W						8			CCR→@aa:32	—	—	—	—	—
	STC EXR,@aa:32	W						8			EXR→@aa:32	—	—	—	—	—
ANDC	ANDC #xx:8,CCR	B	2								CCR∧#xx:8→CCR	↕	↕	↕		
	ANDC #xx:8,EXR	B	4								EXR∧#xx:8→EXR	—	—	—	—	—
ORC	ORC #xx:8,CCR	B	2								CCR∨#xx:8→CCR	↕	↕	↕		
	ORC #xx:8,EXR	B	4								EXR∨#xx:8→EXR	—	—	—	—	—
XORC	XORC #xx:8,CCR	B	2								CCR⊕#xx:8→CCR	↕	↕	↕		
	XORC #xx:8,EXR	B	4								EXR⊕#xx:8→EXR	—	—	—	—	—
NOP	NOP	—								2	PC←PC+2	—	—	—	—	—

											Repeat @ER5→@ER6 ER5+1→ER5 ER6+1→ER6 R4L-1→R4L Until R4L=0 else next;					
	EEPMOV.W	—								4	if R4≠0 Repeat @ER5→@ER6 ER5+1→ER5 ER6+1→ER6 R4-1→R4 Until R4=0 else next;	—	—	—	—	

Notes: 1. The number of states is the number of states required for execution when th
instruction and its operands are located in on-chip memory.

2. n is the initial value set in R4L or R4.

3. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS in

4. Only registers ER0 to ER6 should be used when using the STM/LDM instruc

[1] 7 states when the number of saved/restored registers is 2, 9 states when 3,
states when 4.

[2] Cannot be used with this LSI.

[3] Set to 1 when there is a carry from or borrow to bit 11; otherwise cleared to 0

[4] Set to 1 when there is a carry from or borrow to bit 27; otherwise cleared to 0

[5] If the result is zero, the previous value of the flag is retained; otherwise the fl
cleared to 0.

[6] Set to 1 if the divisor is negative; otherwise cleared to 0.

[7] Set to 1 if the divisor is zero; otherwise cleared to 0.

[8] Set to 1 if the quotient is negative; otherwise cleared to 0.

[9] When EXR is valid, the number of states is increased by 1.

Instruction	Mnemonic	Size	Instruction Format								
			1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8th Byte	9
ADD	ADD.B #xx:8,Rd	B	8 rd	IMM							
	ADD.B Rs,Rd	B	0 8	rs rd							
	ADD.W #xx:16,Rd	W	7 9	1 rd	IMM						
	ADD.W Rs,Rd	W	0 9	rs rd							
	ADD.L #xx:32,ERd	L	7 A	1 0 erd		IMM					
	ADD.L ERs,ERd	L	0 A	1 ers 0 erd							
ADDS	ADDS #1,ERd	L	0 B	0 0 erd							
	ADDS #2,ERd	L	0 B	8 0 erd							
	ADDS #4,ERd	L	0 B	9 0 erd							
ADDX	ADDX #xx:8,Rd	B	9 rd	IMM							
	ADDX Rs,Rd	B	0 E	rs rd							
AND	AND.B #xx:8,Rd	B	E rd	IMM							
	AND.B Rs,Rd	B	1 6	rs rd							
	AND.W #xx:16,Rd	W	7 9	6 rd		IMM					
	AND.W Rs,Rd	W	6 6	rs rd							
	AND.L #xx:32,ERd	L	7 A	6 0 erd			IMM				
	AND.L ERs,ERd	L	0 1	F 0	6	6	0 ers 0 erd				
ANDC	ANDC #xx:8,CCR	B	0 6	IMM							
	ANDC #xx:8,EXR	B	0 1	4 1	0	6	IMM				
BAND	BAND #xx:3,Rd	B	7 6	0 IMM rd							
	BAND #xx:3,@ERd	B	7 C	0 erd 0	7	6	0 IMM 0				
	BAND #xx:3,@aa:8	B	7 E	abs	7	6	0 IMM 0				
	BAND #xx:3,@aa:16	B	6 A	1 0	abs		7 6	0 IMM 0			
	BAND #xx:3,@aa:32	B	6 A	3 0	abs				7 6	0 IMM 0	
Bcc	BRA d:8 (BT d:8)	—	4 0	disp							
	BRA d:16 (BT d:16)	—	5 8	0 0	disp						
	BRN d:8 (BF d:8)	—	4 1	disp							
	BRN d:16 (BF d:16)	—	5 8	1 0	disp						

RENESAS

Instruction	Mnemonic	Size	Instruction Format							
			1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
Bcc	BHI d:8	—	4 2	disp						
	BHI d:16	—	5 8	2 0	disp					
	BLS d:8	—	4 3	disp						
	BLS d:16	—	5 8	3 0	disp					
	BCC d:8 (BHS d:8)	—	4 4	disp						
	BCC d:16 (BHS d:16)	—	5 8	4 0	disp					
	BCS d:8 (BLO d:8)	—	4 5	disp						
	BCS d:16 (BLO d:16)	—	5 8	5 0	disp					
	BNE d:8	—	4 6	disp						
	BNE d:16	—	5 8	6 0	disp					
	BEQ d:8	—	4 7	disp						
	BEQ d:16	—	5 8	7 0	disp					
	BVC d:8	—	4 8	disp						
	BVC d:16	—	5 8	8 0	disp					
	BVS d:8	—	4 9	disp						
	BVS d:16	—	5 8	9 0	disp					
	BPL d:8	—	4 A	disp						
	BPL d:16	—	5 8	A 0	disp					
	BMI d:8	—	4 B	disp						
	BMI d:16	—	5 8	B 0	disp					
	BGE d:8	—	4 C	disp						
	BGE d:16	—	5 8	C 0	disp					
	BLT d:8	—	4 D	disp						
	BLT d:16	—	5 8	D 0	disp					
	BGT d:8	—	4 E	disp						
	BGT d:16	—	5 8	E 0	disp					
	BLE d:8	—	4 F	disp						
	BLE d:16	—	5 8	F 0	disp					

Instruction	Mnemonic	Size	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
BCLR	BCLR #xx:3,Rd	B	7 2	0 IMM rd						
	BCLR #xx:3,@ERd	B	7 D	0 erd 0	7 2	0 IMM 0				
	BCLR #xx:3,@aa:8	B	7 F	abs	7 2	0 IMM 0				
	BCLR #xx:3,@aa:16	B	6 A	1 8	abs		7 2	0 IMM 0		
	BCLR #xx:3,@aa:32	B	6 A	3 8	abs				7 2	0 I
	BCLR Rn,Rd	B	6 2	rn rd						
	BCLR Rn,@ERd	B	7 D	0 erd 0	6 2	rn 0				
	BCLR Rn,@aa:8	B	7 F	abs	6 2	rn 0				
	BCLR Rn,@aa:16	B	6 A	1 8	abs		6 2	rn 0		
	BCLR Rn,@aa:32	B	6 A	3 8	abs				6 2	r
BIAND	BIAND #xx:3,Rd	B	7 6	1 IMM rd						
	BIAND #xx:3,@ERd	B	7 C	0 erd 0	7 6	1 IMM 0				
	BIAND #xx:3,@aa:8	B	7 E	abs	7 6	1 IMM 0				
	BIAND #xx:3,@aa:16	B	6 A	1 0	abs		7 6	1 IMM 0		
	BIAND #xx:3,@aa:32	B	6 A	3 0	abs				7 6	1 I
BILD	BILD #xx:3,Rd	B	7 7	1 IMM rd						
	BILD #xx:3,@ERd	B	7 C	0 erd 0	7 7	1 IMM 0				
	BILD #xx:3,@aa:8	B	7 E	abs	7 7	1 IMM 0				
	BILD #xx:3,@aa:16	B	6 A	1 0	abs		7 7	1 IMM 0		
	BILD #xx:3,@aa:32	B	6 A	3 0	abs				7 7	1 I
BIOR	BIOR #xx:3,Rd	B	7 4	1 IMM rd						
	BIOR #xx:3,@ERd	B	7 C	0 erd 0	7 4	1 IMM 0				
	BIOR #xx:3,@aa:8	B	7 E	abs	7 4	1 IMM 0				
	BIOR #xx:3,@aa:16	B	6 A	1 0	abs		7 4	1 IMM 0		
	BIOR #xx:3,@aa:32	B	6 A	3 0	abs				7 4	1 I

Instruction	Mnemonic	Size	Instruction Format							
			1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
BIST	BIST #xx:3,Rd	B	6 7	1 IMM rd						
	BIST #xx:3,@ERd	B	7 D	0 erd 0	6 7	1 IMM 0				
	BIST #xx:3,@aa:8	B	7 F	abs	6 7	1 IMM 0				
	BIST #xx:3,@aa:16	B	6 A	1 8	abs		6 7	1 IMM 0		
	BIST #xx:3,@aa:32	B	6 A	3 8	abs				6 7	1 I
BIXOR	BIXOR #xx:3,Rd	B	7 5	1 IMM rd						
	BIXOR #xx:3,@ERd	B	7 C	0 erd 0	7 5	1 IMM 0				
	BIXOR #xx:3,@aa:8	B	7 E	abs	7 5	1 IMM 0				
	BIXOR #xx:3,@aa:16	B	6 A	1 0	abs		7 5	1 IMM 0		
	BIXOR #xx:3,@aa:32	B	6 A	3 0	abs				7 5	1 I
BLD	BLD #xx:3,Rd	B	7 7	0 IMM rd						
	BLD #xx:3,@ERd	B	7 C	0 erd 0	7 7	0 IMM 0				
	BLD #xx:3,@aa:8	B	7 E	abs	7 7	0 IMM 0				
	BLD #xx:3,@aa:16	B	6 A	1 0	abs		7 7	0 IMM 0		
	BLD #xx:3,@aa:32	B	6 A	3 0	abs				7 7	0 I
BNOT	BNOT #xx:3,Rd	B	7 1	0 IMM rd						
	BNOT #xx:3,@ERd	B	7 D	0 erd 0	7 1	0 IMM 0				
	BNOT #xx:3,@aa:8	B	7 F	abs	7 1	0 IMM 0				
	BNOT #xx:3,@aa:16	B	6 A	1 8	abs		7 1	0 IMM 0		
	BNOT #xx:3,@aa:32	B	6 A	3 8	abs				7 1	0 I
	BNOT Rn,Rd	B	6 1	rn rd						
	BNOT Rn,@ERd	B	7 D	0 erd 0	6 1	rn 0				
	BNOT Rn,@aa:8	B	7 F	abs	6 1	rn 0				
	BNOT Rn,@aa:16	B	6 A	1 8	abs		6 1	rn 0		
	BNOT Rn,@aa:32	B	6 A	3 8	abs				6 1	r

Instruction	Mnemonic	Size	Instruction Format																											
			1st Byte				2nd Byte				3rd Byte				4th Byte				5th Byte				6th Byte				7th Byte			
BOR	BOR #xx:3,Rd	B	7	4	0	IMM	rd																							
	BOR #xx:3,@ERd	B	7	C	0	erd	0				7	4	0	IMM	0															
	BOR #xx:3,@aa:8	B	7	E	abs						7	4	0	IMM	0															
	BOR #xx:3,@aa:16	B	6	A	1	0	abs												7	4	0	IMM	0							
	BOR #xx:3,@aa:32	B	6	A	3	0	abs																				7	4	0	I
BSET	BSET #xx:3,Rd	B	7	0	0	IMM	rd																							
	BSET #xx:3,@ERd	B	7	D	0	erd	0				7	0	0	IMM	0															
	BSET #xx:3,@aa:8	B	7	F	abs						7	0	0	IMM	0															
	BSET #xx:3,@aa:16	B	6	A	1	8	abs												7	0	0	IMM	0							
	BSET #xx:3,@aa:32	B	6	A	3	8	abs																				7	0	0	I
	BSET Rn,Rd	B	6	0	rn	rd																								
	BSET Rn,@ERd	B	7	D	0	erd	0				6	0	rn	0																
	BSET Rn,@aa:8	B	7	F	abs						6	0	rn	0																
	BSET Rn,@aa:16	B	6	A	1	8	abs												6	0	rn	0								
	BSET Rn,@aa:32	B	6	A	3	8	abs																				6	0	r	
BSR	BSR d:8	—	5	5	disp																									
	BSR d:16	—	5	C	0	0	disp																							
BST	BST #xx:3,Rd	B	6	7	0	IMM	rd																							
	BST #xx:3,@ERd	B	7	D	0	erd	0				6	7	0	IMM	0															
	BST #xx:3,@aa:8	B	7	F	abs						6	7	0	IMM	0															
	BST #xx:3,@aa:16	B	6	A	1	8	abs												6	7	0	IMM	0							
	BST #xx:3,@aa:32	B	6	A	3	8	abs																				6	7	0	I
BTST	BTST #xx:3,Rd	B	7	3	0	IMM	rd																							
	BTST #xx:3,@ERd	B	7	C	0	erd	0				7	3	0	IMM	0															
	BTST #xx:3,@aa:8	B	7	E	abs						7	3	0	IMM	0															
	BTST #xx:3,@aa:16	B	6	A	1	0	abs												7	3	0	IMM	0							
	BTST #xx:3,@aa:32	B	6	A	3	0	abs																				7	3	0	I
	BTST Rn,Rd	B	6	3	rn	rd																								

Instruction	Mnemonic	Size	Instruction Format 1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
BTST	BTST Rn,@aa:8	B	7 E	abs	6 3	rn 0				
	BTST Rn,@aa:16	B	6 A	1 0	abs		6 3	rn 0		
	BTST Rn,@aa:32	B	6 A	3 0	abs				6 3	r
BXOR	BXOR #xx:3,Rd	B	7 5	0 IMM rd						
	BXOR #xx:3,@ERd	B	7 C	0 erd 0	7 5	0 IMM 0				
	BXOR #xx:3,@aa:8	B	7 E	abs	7 5	0 IMM 0				
	BXOR #xx:3,@aa:16	B	6 A	1 0	abs		7 5	0 IMM 0		
	BXOR #xx:3,@aa:32	B	6 A	3 0	abs				7 5	0 I
CLRMAC	CLRMAC	—	Cannot be used with this LSI.							
CMP	CMP.B #xx:8,Rd	B	A rd	IMM						
	CMP.B Rs,Rd	B	1 C	rs rd						
	CMP.W #xx:16,Rd	W	7 9	2 rd	IMM					
	CMP.W Rs,Rd	W	1 D	rs rd						
	CMP.L #xx:32,ERd	L	7 A	2 0 erd	IMM					
	CMP.L ERs,ERd	L	1 F	1 ers 0 erd						
DAA	DAA Rd	B	0 F	0 rd						
DAS	DAS Rd	B	1 F	0 rd						
DEC	DEC.B Rd	B	1 A	0 rd						
	DEC.W #1,Rd	W	1 B	5 rd						
	DEC.W #2,Rd	W	1 B	D rd						
	DEC.L #1,ERd	L	1 B	7 0 erd						
	DEC.L #2,ERd	L	1 B	F 0 erd						
DIVXS	DIVXS.B Rs,Rd	B	0 1	D 0	5 1	rs rd				
	DIVXS.W Rs,ERd	W	0 1	D 0	5 3	rs 0 erd				
DIVXU	DIVXU.B Rs,Rd	B	5 1	rs rd						
	DIVXU.W Rs,ERd	W	5 3	rs 0 erd						
EEPMOV	EEPMOV.B	—	7 B	5 C	5 9	8 F				
	EEPMOV.W	—	7 B	D 4	5 9	8 F				

Instruc-tion	Mnemonic	Size	Instruction Format																													
			1st Byte				2nd Byte				3rd Byte				4th Byte				5th Byte				6th Byte				7th Byte				8	
EXTS	EXTS.W Rd	W	1	7	D	rd																										
	EXTS.L ERd	L	1	7	F	0	erd																									
EXTU	EXTU.W Rd	W	1	7	5	rd																										
	EXTU.L ERd	L	1	7	7	0	erd																									
INC	INC.B Rd	B	0	A	0	rd																										
	INC.W #1,Rd	W	0	B	5	rd																										
	INC.W #2,Rd	W	0	B	D	rd																										
	INC.L #1,ERd	L	0	B	7	0	erd																									
	INC.L #2,ERd	L	0	B	F	0	erd																									
JMP	JMP @ERn	—	5	9	0	ern	0																									
	JMP @aa:24	—	5	A	abs																											
	JMP @@aa:8	—	5	B	abs																											
JSR	JSR @ERn	—	5	D	0	ern	0																									
	JSR @aa:24	—	5	E	abs																											
	JSR @@aa:8	—	5	F	abs																											
LDC	LDC #xx:8,CCR	B	0	7	IMM																											
	LDC #xx:8,EXR	B	0	1	4	1	0	7	IMM																							
	LDC Rs,CCR	B	0	3	0	rs																										
	LDC Rs,EXR	B	0	3	1	rs																										
	LDC @ERs,CCR	W	0	1	4	0	6	9	0	ers	0																					
	LDC @ERs,EXR	W	0	1	4	1	6	9	0	ers	0																					
	LDC @(d:16,ERs),CCR	W	0	1	4	0	6	F	0	ers	0	disp																				
	LDC @(d:16,ERs),EXR	W	0	1	4	1	6	F	0	ers	0	disp																				
	LDC @(d:32,ERs),CCR	W	0	1	4	0	7	8	0	ers	0	6	B	2	0																	
	LDC @(d:32,ERs),EXR	W	0	1	4	1	7	8	0	ers	0	6	B	2	0																	
	LDC @ERs+,CCR	W	0	1	4	0	6	D	0	ers	0																					
	LDC @ERs+,EXR	W	0	1	4	1	6	D	0	ers	0																					
	LDC @aa:16,CCR	W	0	1	4	0	6	B	0	0	abs																					

Instruction	Mnemonic	Size	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
LDC	LDC @aa:32,CCR	W	0 1	4 0	6 B	2 0	abs			
	LDC @aa:32,EXR	W	0 1	4 1	6 B	2 0	abs			
LDM*3	LDM.L @SP+, (ERn-ERn+1)	L	0 1	1 0	6 D	7 0 ern+1				
	LDM.L @SP+, (ERn-ERn+2)	L	0 1	2 0	6 D	7 0 ern+2				
	LDM.L @SP+, (ERn-ERn+3)	L	0 1	3 0	6 D	7 0 ern+3				
LDMAC	LDMAC ERs,MACH	L	Cannot be used with this LSI.							
	LDMAC ERs,MACL	L								
MAC	MAC @ERn+,@ERm+	—								
MOV	MOV.B #xx:8,Rd	B	F rd	IMM						
	MOV.B Rs,Rd	B	0 C	rs rd						
	MOV.B @ERs,Rd	B	6 8	0 ers rd						
	MOV.B @(d:16,ERs),Rd	B	6 E	0 ers rd	disp					
	MOV.B @(d:32,ERs),Rd	B	7 8	0 ers 0	6 A	2 rd	disp			
	MOV.B @ERs+,Rd	B	6 C	0 ers rd						
	MOV.B @aa:8,Rd	B	2 rd	abs						
	MOV.B @aa:16,Rd	B	6 A	0 rd	abs					
	MOV.B @aa:32,Rd	B	6 A	2 rd	abs					
	MOV.B Rs,@ERd	B	6 8	1 erd rs						
	MOV.B Rs,@(d:16,ERd)	B	6 E	1 erd rs	disp					
	MOV.B Rs,@(d:32,ERd)	B	7 8	0 erd 0	6 A	A rs	disp			
	MOV.B Rs,@-ERd	B	6 C	1 erd rs						
	MOV.B Rs,@aa:8	B	3 rs	abs						
	MOV.B Rs,@aa:16	B	6 A	8 rs	abs					
	MOV.B Rs,@aa:32	B	6 A	A rs	abs					
	MOV.W #xx:16,Rd	W	7 9	0 rd	IMM					
	MOV.W Rs,Rd	W	0 D	rs rd						
	MOV.W @ERs,Rd	W	6 9	0 ers rd						
	MOV.W @(d:16,ERs),Rd	W	6 F	0 ers rd	disp					
	MOV.W @(d:32,ERs),Rd	W	7 8	0 ers 0	6 B	2 rd	disp			

Instruction	Mnemonic	Size	Instruction Format 1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
MOV	MOV.W @ERs+,Rd	W	6 D	0 ers rd						
	MOV.W @aa:16,Rd	W	6 B	0 rd	abs					
	MOV.W @aa:32,Rd	W	6 B	2 rd	abs					
	MOV.W Rs,@ERd	W	6 9	1 erd rs						
	MOV.W Rs,@(d:16,ERd)	W	6 F	1 erd rs	disp					
	MOV.W Rs,@(d:32,ERd)	W	7 8	0 erd 0	6 B	A rs	disp			
	MOV.W Rs,@-ERd	W	6 D	1 erd rs						
	MOV.W Rs,@aa:16	W	6 B	8 rs	abs					
	MOV.W Rs,@aa:32	W	6 B	A rs	abs					
	MOV.L #xx:32,Rd	L	7 A	0 0 erd	IMM					
	MOV.L ERs,ERd	L	0 F	1 ers 0 erd						
	MOV.L @ERs,ERd	L	0 1	0 0	6 9	0 ers 0 erd				
	MOV.L @(d:16,ERs),ERd	L	0 1	0 0	6 F	0 ers 0 erd	disp			
	MOV.L @(d:32,ERs),ERd	L	0 1	0 0	7 8	0 ers 0	6 B	2 0 erd		
	MOV.L @ERs+,ERd	L	0 1	0 0	6 D	0 ers 0 erd				
	MOV.L @aa:16 ,ERd	L	0 1	0 0	6 B	0 0 erd	abs			
	MOV.L @aa:32 ,ERd	L	0 1	0 0	6 B	2 0 erd	abs			
	MOV.L ERs,@ERd	L	0 1	0 0	6 9	1 erd 0 ers				
	MOV.L ERs,@(d:16,ERd)	L	0 1	0 0	6 F	1 erd 0 ers	disp			
	MOV.L ERs,@(d:32,ERd)*1	L	0 1	0 0	7 8	0 erd 0	6 B	A 0 ers		
	MOV.L ERs,@-ERd	L	0 1	0 0	6 D	1 erd 0 ers				
	MOV.L ERs,@aa:16	L	0 1	0 0	6 B	8 0 ers	abs			
	MOV.L ERs,@aa:32	L	0 1	0 0	6 B	A 0 ers	abs			
MOVFPE	MOVFPE @aa:16,Rd	B	Cannot be used with this LSI.							
MOVTPE	MOVTPE Rs,@aa:16	B								
MULXS	MULXS.B Rs,Rd	B	0 1	C 0	5 0	rs rd				
	MULXS.W Rs,ERd	W	0 1	C 0	5 2	rs 0 erd				
MULXU	MULXU.B Rs,Rd	B	5 0	rs rd						

Instruction	Mnemonic	Size	Instruction Format							
			1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
NEG	NEG.B Rd	B	1 7	8 rd						
	NEG.W Rd	W	1 7	9 rd						
	NEG.L ERd	L	1 7	B 0 erd						
NOP	NOP	—	0 0	0 0						
NOT	NOT.B Rd	B	1 7	0 rd						
	NOT.W Rd	W	1 7	1 rd						
	NOT.L ERd	L	1 7	3 0 erd						
OR	OR.B #xx:8,Rd	B	C rd	IMM						
	OR.B Rs,Rd	B	1 4	rs rd						
	OR.W #xx:16,Rd	W	7 9	4 rd	IMM					
	OR.W Rs,Rd	W	6 4	rs rd						
	OR.L #xx:32,ERd	L	7 A	4 0 erd	IMM					
	OR.L ERs,ERd	L	0 1	F 0	6 4	0 ers 0 erd				
ORC	ORC #xx:8,CCR	B	0 4	IMM						
	ORC #xx:8,EXR	B	0 1	4 1	0 4	IMM				
POP	POP.W Rn	W	6 D	7 rn						
	POP.L ERn	L	0 1	0 0	6 D	7 0 ern				
PUSH	PUSH.W Rn	W	6 D	F rn						
	PUSH.L ERn	L	0 1	0 0	6 D	F 0 ern				
ROTL	ROTL.B Rd	B	1 2	8 rd						
	ROTL.B #2, Rd	B	1 2	C rd						
	ROTL.W Rd	W	1 2	9 rd						
	ROTL.W #2, Rd	W	1 2	D rd						
	ROTL.L ERd	L	1 2	B 0 erd						
	ROTL.L #2, ERd	L	1 2	F 0 erd						

Instruction	Mnemonic	Size	Instruction Format							
			1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
ROTR	ROTR.B Rd	B	1 3	8 rd						
	ROTR.B #2, Rd	B	1 3	C rd						
	ROTR.W Rd	W	1 3	9 rd						
	ROTR.W #2, Rd	W	1 3	D rd						
	ROTR.L ERd	L	1 3	B 0: erd						
	ROTR.L #2, ERd	L	1 3	F 0: erd						
ROTXL	ROTXL.B Rd	B	1 2	0 rd						
	ROTXL.B #2, Rd	B	1 2	4 rd						
	ROTXL.W Rd	W	1 2	1 rd						
	ROTXL.W #2, Rd	W	1 2	5 rd						
	ROTXL.L ERd	L	1 2	3 0: erd						
	ROTXL.L #2, ERd	L	1 2	7 0: erd						
ROTXR	ROTXR.B Rd	B	1 3	0 rd						
	ROTXR.B #2, Rd	B	1 3	4 rd						
	ROTXR.W Rd	W	1 3	1 rd						
	ROTXR.W #2, Rd	W	1 3	5 rd						
	ROTXR.L ERd	L	1 3	3 0: erd						
	ROTXR.L #2, ERd	L	1 3	7 0: erd						
RTE	RTE	—	5 6	7 0						
RTS	RTS	—	5 4	7 0						
SHAL	SHAL.B Rd	B	1 0	8 rd						
	SHAL.B #2, Rd	B	1 0	C rd						
	SHAL.W Rd	W	1 0	9 rd						
	SHAL.W #2, Rd	W	1 0	D rd						
	SHAL.L ERd	L	1 0	B 0: erd						
	SHAL.L #2, ERd	L	1 0	F 0: erd						

RENESAS

Instruction	Mnemonic	Size	Instruction Format: 1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8
SHAR	SHAR.B Rd	B	1 1	8 rd						
	SHAR.B #2, Rd	B	1 1	C rd						
	SHAR.W Rd	W	1 1	9 rd						
	SHAR.W #2, Rd	W	1 1	D rd						
	SHAR.L ERd	L	1 1	B 0 erd						
	SHAR.L #2, ERd	L	1 1	F 0 erd						
SHLL	SHLL.B Rd	B	1 0	0 rd						
	SHLL.B #2, Rd	B	1 0	4 rd						
	SHLL.W Rd	W	1 0	1 rd						
	SHLL.W #2, Rd	W	1 0	5 rd						
	SHLL.L ERd	L	1 0	3 0 erd						
	SHLL.L #2, ERd	L	1 0	7 0 erd						
SHLR	SHLR.B Rd	B	1 1	0 rd						
	SHLR.B #2, Rd	B	1 1	4 rd						
	SHLR.W Rd	W	1 1	1 rd						
	SHLR.W #2, Rd	W	1 1	5 rd						
	SHLR.L ERd	L	1 1	3 0 erd						
	SHLR.L #2, ERd	L	1 1	7 0 erd						
SLEEP	SLEEP	—	0 1	8 0						
STC	STC.B CCR,Rd	B	0 2	0 rd						
	STC.B EXR,Rd	B	0 2	1 rd						
	STC.W CCR,@ERd	W	0 1	4 0	6 9	1 erd 0				
	STC.W EXR,@ERd	W	0 1	4 1	6 9	1 erd 0				
	STC.W CCR,@(d:16,ERd)	W	0 1	4 0	6 F	1 erd 0	disp			
	STC.W EXR,@(d:16,ERd)	W	0 1	4 1	6 F	1 erd 0	disp			
	STC.W CCR,@(d:32,ERd)	W	0 1	4 0	7 8	0 erd 0	6 B	A 0		
	STC.W EXR,@(d:32,ERd)	W	0 1	4 1	7 8	0 erd 0	6 B	A 0		
	STC.W CCR,@-ERd	W	0 1	4 0	6 D	1 erd 0				
	STC.W EXR,@-ERd	W	0 1	4 1	6 D	1 erd 0				

RENESAS

Instruction	Mnemonic	Size	Instruction Format																			
			1st Byte		2nd Byte		3rd Byte		4th Byte		5th Byte		6th Byte		7th Byte		8					
STC	STC.W CCR,@aa:16	W	0	1	4	0	6	B	8	0	abs											
	STC.W EXR,@aa:16	W	0	1	4	1	6	B	8	0	abs											
	STC.W CCR,@aa:32	W	0	1	4	0	6	B	A	0	abs											
	STC.W EXR,@aa:32	W	0	1	4	1	6	B	A	0	abs											
STM*3	STM.L (ERn-ERn+1), @-SP	L	0	1	1	0	6	D	F	0	ern											
	STM.L (ERn-ERn+2), @-SP	L	0	1	2	0	6	D	F	0	ern											
	STM.L (ERn-ERn+3), @-SP	L	0	1	3	0	6	D	F	0	ern											
STMAC	STMAC MACH,ERd	L	Cannot be used with this LSI.																			
	STMAC MACL,ERd	L																				
SUB	SUB.B Rs,Rd	B	1	8	rs		rd															
	SUB.W #xx:16,Rd	W	7	9	3		rd		IMM													
	SUB.W Rs,Rd	W	1	9	rs		rd															
	SUB.L #xx:32,ERd	L	7	A	3	0	erd		IMM													
	SUB.L ERs,ERd	L	1	A	1	ers	0	erd														
SUBS	SUBS #1,ERd	L	1	B	0	0	erd															
	SUBS #2,ERd	L	1	B	8	0	erd															
	SUBS #4,ERd	L	1	B	9	0	erd															
SUBX	SUBX #xx:8,Rd	B	B	rd	IMM																	
	SUBX Rs,Rd	B	1	E	rs		rd															
TAS	TAS @ERd*2	B	0	1	E	0	7	B	0	erd	C											
TRAPA	TRAPA #x:2	—	5	7	00	IMM	0															
XOR	XOR.B #xx:8,Rd	B	D	rd	IMM																	
	XOR.B Rs,Rd	B	1	5	rs		rd															
	XOR.W #xx:16,Rd	W	7	9	5		rd		IMM													
	XOR.W Rs,Rd	W	6	5	rs		rd															
	XOR.L #xx:32,ERd	L	7	A	5	0	erd		IMM													
	XOR.L ERs,ERd	L	0	1	F	0	6	5	0	ers	0	erd										

Instruc-tion	Mnemonic	Size	Instruction Format							
			1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	
XORC	XORC #xx:8,CCR	B	0 5	IMM						
	XORC #xx:8,EXR	B	0 1	4 1	0 5	IMM				

Legend:

IMM:	Immediate data (2, 3, 8, 16, or 32 bits)
abs:	Absolute address (8, 16, 24, or 32 bits)
disp:	Displacement (8, 16, or 32 bits)
rs, rd, rn:	Register field (4 bits, indicating an 8-bit or 16-bit register. rs, rd, and rn correspond to operand formats
ers, erd, ern, erm:	Register field (3 bits, indicating an address register or 32-bit register. ers, erd, ern, and erm correspon ERn, and ERm, respectively.)

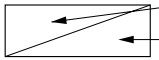
Notes: 1. Bit 7 of the 4th byte of the MOV.L ERs, @ (d:32, ERd) instruction can be either 0 or 1.
2. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS instruction.
3. Only registers ER0 to ER6 should be used when using the STM/LDM instruction.

The correspondence between register fields and general registers is shown in the following table.

Address Registers 32-Bit Registers		16-Bit Register		8-Bit Register	
Register Field	General Register	Register Field	General Register	Register Field	General Register
000	ER0	0000	R0	0000	R0H
001	ER1	0001	R1	0001	R1H
•	•	•	•	•	•
•	•	•	•	•	•
•	•	•	•	•	•
111	ER7	0111	R7	0111	R7H
		1000	E0	1000	R0L
		1001	E1	1001	R1L
		•	•	•	•
		•	•	•	•
		•	•	•	•
		1111	E7	1111	R7L

Instruction code:

1st byte		2nd byte	
AH	AL	BH	BL



Instruction when most significant bit of BH is 0.

Instruction when most significant bit of BH is 1.

AH \ AL	0	1	2	3	4	5	6	7	8	9	A	B	C	D	
0	NOP	Table A.3 (2)	STC / STMAC*	LDC / LDMAC*	ORC	XORC	ANDC	LDC	ADD		Table A.3 (2)	Table A.3 (2)	MOV		A
1	Table A.3 (2)	Table A.3 (2)	Table A.3 (2)	Table A.3 (2)	OR	XOR	AND	Table A.3 (2)	SUB		Table A.3 (2)	Table A.3 (2)	CMP		S
2	MOV.B														
3															
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	B
5	MULXU	DIVXU	MULXU	DIVXU	RTS	BSR	RTE	TRAPA	Table A.3 (2)	JMP			BSR		J
6	BSET	BNOT	BCLR	BTST	OR	XOR	AND	BST / BIST	MOV		Table A.3 (2)	MOV			
7					BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD	MOV	Table A.3 (2)	Table A.3 (2)	EEPMOV	Table A.3		
8	ADD														
9	ADDX														
A	CMP														
B	SUBX														
C	OR														
D	XOR														
E	AND														
F	MOV														

Note: * Cannot be used with this LSI.

Instruction code:

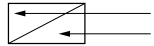
1st byte		2nd byte	
AH	AL	BH	BL

AH AL \ BH	0	1	2	3	4	5	6	7	8	9	A	B	C
01	MOV	LDM / STM			LDC / STC		MAC*		SLEEP		CLRMAC*		Table A.3 (3)
0A	INC								ADD				
0B	ADDS					INC		INC	ADDS				
0F	DAA								MOV				
10	SHLL			SHLL				SHLL	SHAL			SHAL	
11	SHLR			SHLR				SHLR	SHAR			SHAR	
12	ROTXL			ROTXL				ROTXL	ROTL			ROTL	
13	ROTXR			ROTXR				ROTXR	ROTR			ROTR	
17	NOT			NOT		EXTU		EXTU	NEG			NEG	
1A	DEC								SUB				
1B	SUBS					DEC		DEC	SUBS				
1F	DAS								CMP				
58	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE
6A	MOV	Table A.3 (4)	MOV	Table A.3 (4)	MOVFPE*				MOV		MOV		MOVTPE
79	MOV	ADD	CMP	SUB	OR	XOR	AND						
7A	MOV	ADD	CMP	SUB	OR	XOR	AND						

Note: * Cannot be used with this LSI.

Instruction code:

1st byte		2nd byte		3rd byte		4th byte	
AH	AL	BH	BL	CH	CL	DH	DL



AH AL BH BL CH \ CL	0	1	2	3	4	5	6	7	8	9	A	B	C
01C05	MULXS		MULXS										
01D05		DIVXS		DIVXS									
01F06					OR	XOR	AND						
7Cr06*1				BTST									
7Cr07*1				BTST	BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD					
7Dr06*1	BSET	BNOT	BCLR					BST / BIST					
7Dr07*1	BSET	BNOT	BCLR										
7Eaa6*2				BTST									
7Eaa7*2				BTST	BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD					
7Faa6*2	BSET	BNOT	BCLR					BST / BIST					
7Faa7*2	BSET	BNOT	BCLR										

Notes: 1. r is the register specification field.
2. aa is the absolute address specification.

Instruction code:

1st byte		2nd byte		3rd byte		4th byte		5th byte		6th byte	
AH	AL	BH	BL	CH	CL	DH	DL	EH	EL	FH	FL

Instruction whe
Instruction whe

AHALBHBLCHCLDHDLEH \ EL	0	1	2	3	4	5	6	7	8	9	A	B	C
6A10aaaa6*				BTST									
6A10aaaa7*					BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD					
6A18aaaa6*	BSET	BNOT	BCLR					BST / BIST					
6A18aaaa7*													

Instruction code:

1st byte		2nd byte		3rd byte		4th byte		5th byte		6th byte		7th byte		8th b
AH	AL	BH	BL	CH	CL	DH	DL	EH	EL	FH	FL	GH	GL	HH

Indicates case
Indicates case

AHALBHBL ... FHFLGH \ GL	0	1	2	3	4	5	6	7	8	9	A	B	C
6A30aaaaaaaa6*				BTST									
6A30aaaaaaaa7*					BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD					
6A38aaaaaaaa6*	BSET	BNOT	BCLR					BST / BIST					
6A38aaaaaaaa7*													

Note: * aa is the absolute address specification

Examples of Calculation of Number of States Required for Execution

Examples: Advanced mode, stack located in external address space, on-chip supportin
accessed in two states with 8-bit bus width, external devices accessed in three states w
state and 16-bit bus width.

1. BSET #0,@FFFFC7:8

   From table A.5,
   I = L = 2 and J = K = M = N = 0

   From table A.4,
   $S_I = 4$ and $S_L = 2$

   Number of states = $2 \times 4 + 2 \times 2 = 12$

2. JSR @@30

   From table A.5,
   I = J = K = 2 and L = M = N = 0

   From table A.4,
   $S_I = S_J = S_K = 4$

   Number of states = $2 \times 4 + 2 \times 4 + 2 \times 4 = 24$

Instruction fetch	S_I	2	4	2	4	6 + 2m	2
Branch address fetch	S_J						
Stack operation	S_K						
Byte data access	S_L		2		2	3 + m	
Word data access	S_M		4		4	6 + 2m	
Internal operation	S_N	1	1	1	1	1	1

Legend:

m: Number of wait states inserted into external device access

	ADD.L #xx:32,ERd	3		
	ADD.L ERs,ERd	1		
ADDS	ADDS #1/2/4,ERd	1		
ADDX	ADDX #xx:8,Rd	1		
	ADDX Rs,Rd	1		
AND	AND.B #xx:8,Rd	1		
	AND.B Rs,Rd	1		
	AND.W #xx:16,Rd	2		
	AND.W Rs,Rd	1		
	AND.L #xx:32,ERd	3		
	AND.L ERs,ERd	2		
ANDC	ANDC #xx:8,CCR	1		
	ANDC #xx:8,EXR	2		
BAND	BAND #xx:3,Rd	1		
	BAND #xx:3,@ERd	2		1
	BAND #xx:3,@aa:8	2		1
	BAND #xx:3,@aa:16	3		1
	BAND #xx:3,@aa:32	4		1
Bcc	BRA d:8 (BT d:8)	2		
	BRN d:8 (BF d:8)	2		
	BHI d:8	2		
	BLS d:8	2		
	BCC d:8 (BHS d:8)	2		
	BCS d:8 (BLO d:8)	2		
	BNE d:8	2		
	BEQ d:8	2		
	BVC d:8	2		
	BVS d:8	2		
	BPL d:8	2		

	BRA d:16 (BT d:16)	2		
	BRN d:16 (BF d:16)	2		
	BHI d:16	2		
	BLS d:16	2		
	BCC d:16 (BHS d:16)	2		
	BCS d:16 (BLO d:16)	2		
	BNE d:16	2		
	BEQ d:16	2		
	BVC d:16	2		
	BVS d:16	2		
	BPL d:16	2		
	BMI d:16	2		
	BGE d:16	2		
	BLT d:16	2		
	BGT d:16	2		
	BLE d:16	2		
BCLR	BCLR #xx:3,Rd	1		
	BCLR #xx:3,@ERd	2		2
	BCLR #xx:3,@aa:8	2		2
	BCLR #xx:3,@aa:16	3		2
	BCLR #xx:3,@aa:32	4		2
	BCLR Rn,Rd	1		
	BCLR Rn,@ERd	2		2
	BCLR Rn,@aa:8	2		2
	BCLR Rn,@aa:16	3		2
	BCLR Rn,@aa:32	4		2
BIAND	BIAND #xx:3,Rd	1		
	BIAND #xx:3,@ERd	2		1
	BIAND #xx:3,@aa:8	2		1
	BIAND #xx:3,@aa:16	3		1
	BIAND #xx:3,@aa:32	4		1

BIOR	BIOR #xx:8,Rd	1	
	BIOR #xx:8,@ERd	2	1
	BIOR #xx:8,@aa:8	2	1
	BIOR #xx:8,@aa:16	3	1
	BIOR #xx:8,@aa:32	4	1
BIST	BIST #xx:3,Rd	1	
	BIST #xx:3,@ERd	2	2
	BIST #xx:3,@aa:8	2	2
	BIST #xx:3,@aa:16	3	2
	BIST #xx:3,@aa:32	4	2
BIXOR	BIXOR #xx:3,Rd	1	
	BIXOR #xx:3,@ERd	2	1
	BIXOR #xx:3,@aa:8	2	1
	BIXOR #xx:3,@aa:16	3	1
	BIXOR #xx:3,@aa:32	4	1
BLD	BLD #xx:3,Rd	1	
	BLD #xx:3,@ERd	2	1
	BLD #xx:3,@aa:8	2	1
	BLD #xx:3,@aa:16	3	1
	BLD #xx:3,@aa:32	4	1
BNOT	BNOT #xx:3,Rd	1	
	BNOT #xx:3,@ERd	2	2
	BNOT #xx:3,@aa:8	2	2
	BNOT #xx:3,@aa:16	3	2
	BNOT #xx:3,@aa:32	4	2
	BNOT Rn,Rd	1	
	BNOT Rn,@ERd	2	2
	BNOT Rn,@aa:8	2	2
	BNOT Rn,@aa:16	3	2
	BNOT Rn,@aa:32	4	2

BSET	BSET #xx:3,Rd		1		
	BSET #xx:3,@ERd		2		2
	BSET #xx:3,@aa:8		2		2
	BSET #xx:3,@aa:16		3		2
	BSET #xx:3,@aa:32		4		2
	BSET Rn,Rd		1		
	BSET Rn,@ERd		2		2
	BSET Rn,@aa:8		2		2
	BSET Rn,@aa:16		3		2
	BSET Rn,@aa:32		4		2
BSR	BSR d:8	Normal	2	1	
		Advanced	2	2	
	BSR d:16	Normal	2	1	
		Advanced	2	2	
BST	BST #xx:3,Rd		1		
	BST #xx:3,@ERd		2		2
	BST #xx:3,@aa:8		2		2
	BST #xx:3,@aa:16		3		2
	BST #xx:3,@aa:32		4		2
BTST	BTST #xx:3,Rd		1		
	BTST #xx:3,@ERd		2		1
	BTST #xx:3,@aa:8		2		1
	BTST #xx:3,@aa:16		3		1
	BTST #xx:3,@aa:32		4		1
	BTST Rn,Rd		1		
	BTST Rn,@ERd		2		1
	BTST Rn,@aa:8		2		1
	BTST Rn,@aa:16		3		1
	BTST Rn,@aa:32		4		1

CLRMAC	CLRMAC		Cannot be used with this LSI.		
CMP	CMP.B #xx:8,Rd		1		
	CMP.B Rs,Rd		1		
	CMP.W #xx:16,Rd		2		
	CMP.W Rs,Rd		1		
	CMP.L #xx:32,ERd		3		
	CMP.L ERs,ERd		1		
DAA	DAA Rd		1		
DAS	DAS Rd		1		
DEC	DEC.B Rd		1		
	DEC.W #1/2,Rd		1		
	DEC.L #1/2,ERd		1		
DIVXS	DIVXS.B Rs,Rd		2		
	DIVXS.W Rs,ERd		2		
DIVXU	DIVXU.B Rs,Rd		1		
	DIVXU.W Rs,ERd		1		
EEPMOV	EEPMOV.B		2		2n+2*2
	EEPMOV.W		2		2n+2*2
EXTS	EXTS.W Rd		1		
	EXTS.L ERd		1		
EXTU	EXTU.W Rd		1		
	EXTU.L ERd		1		
INC	INC.B Rd		1		
	INC.W #1/2,Rd		1		
	INC.L #1/2,ERd		1		
JMP	JMP @ERn		2		
	JMP @aa:24		2		
	JMP @@aa:8	Normal	2	1	
		Advanced	2	2	

	Advanced	2	2	2		
LDC	LDC #xx:8,CCR	1				
	LDC #xx:8,EXR	2				
	LDC Rs,CCR	1				
	LDC Rs,EXR	1				
	LDC @ERs,CCR	2				1
	LDC @ERs,EXR	2				1
	LDC @(d:16,ERs),CCR	3				1
	LDC @(d:16,ERs),EXR	3				1
	LDC @(d:32,ERs),CCR	5				1
	LDC @(d:32,ERs),EXR	5				1
	LDC @ERs+,CCR	2				1
	LDC @ERs+,EXR	2				1
	LDC @aa:16,CCR	3				1
	LDC @aa:16,EXR	3				1
	LDC @aa:32,CCR	4				1
	LDC @aa:32,EXR	4				1
LDM*4	LDM.L @SP+, (ERn-ERn+1)	2		4		
	LDM.L @SP+, (ERn-ERn+2)	2		6		
	LDM.L @SP+, (ERn-ERn+3)	2		8		
LDMAC	LDMAC ERs, MACH	Cannot be used with this LSI.				
	LDMAC ERs, MACL					
MAC	MAC @ERn+, @ERm+					
MOV	MOV.B #xx:8,Rd	1				
	MOV.B Rs,Rd	1				
	MOV.B @ERs,Rd	1			1	
	MOV.B @(d:16,ERs),Rd	2			1	
	MOV.B @(d:32,ERs),Rd	4			1	
	MOV.B @ERs+,Rd	1			1	
	MOV.B @aa:8,Rd	1			1	
	MOV.B @aa:16,Rd	2			1	

MOV.B Rs,@aa:8	1		1	
MOV.B Rs,@aa:16	2		1	
MOV.B Rs,@aa:32	3		1	
MOV.W #xx:16,Rd	2			
MOV.W Rs,Rd	1			
MOV.W @ERs,Rd	1			1
MOV.W @(d:16,ERs),Rd	2			1
MOV.W @(d:32,ERs),Rd	4			1
MOV.W @ERs+,Rd	1			1
MOV.W @aa:16,Rd	2			1
MOV.W @aa:32,Rd	3			1
MOV.W Rs,@ERd	1			1
MOV.W Rs,@(d:16,ERd)	2			1
MOV.W Rs,@(d:32,ERd)	4			1
MOV.W Rs,@-ERd	1			1
MOV.W Rs,@aa:16	2			1
MOV.W Rs,@aa:32	3			1
MOV.L #xx:32,ERd	3			
MOV.L ERs,ERd	1			
MOV.L @ERs,ERd	2			2
MOV.L @(d:16,ERs),ERd	3			2
MOV.L @(d:32,ERs),ERd	5			2
MOV.L @ERs+,ERd	2			2
MOV.L @aa:16,ERd	3			2
MOV.L @aa:32,ERd	4			2
MOV.L ERs,@ERd	2			2
MOV.L ERs,@(d:16,ERd)	3			2
MOV.L ERs,@(d:32,ERd)	5			2
MOV.L ERs,@-ERd	2			2
MOV.L ERs,@aa:16	3			2
MOV.L ERs,@aa:32	4			2

	MULXU.W Rs,ERd	1		
NEG	NEG.B Rd	1		
	NEG.W Rd	1		
	NEG.L ERd	1		
NOP	NOP	1		
NOT	NOT.B Rd	1		
	NOT.W Rd	1		
	NOT.L ERd	1		
OR	OR.B #xx:8,Rd	1		
	OR.B Rs,Rd	1		
	OR.W #xx:16,Rd	2		
	OR.W Rs,Rd	1		
	OR.L #xx:32,ERd	3		
	OR.L ERs,ERd	2		
ORC	ORC #xx:8,CCR	1		
	ORC #xx:8,EXR	2		
POP	POP.W Rn	1		1
	POP.L ERn	2		2
PUSH	PUSH.W Rn	1		1
	PUSH.L ERn	2		2
ROTL	ROTL.B Rd	1		
	ROTL.B #2,Rd	1		
	ROTL.W Rd	1		
	ROTL.W #2,Rd	1		
	ROTL.L ERd	1		
	ROTL.L #2,ERd	1		

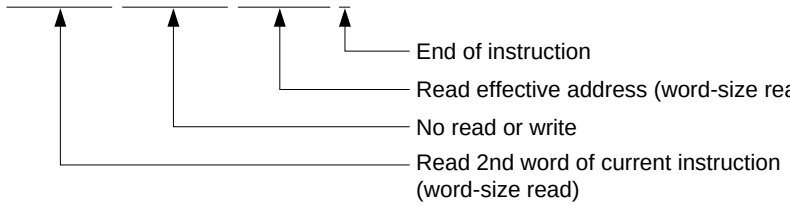
	ROTR.L #2,ERd		1	
ROTXL	ROTXL.B Rd		1	
	ROTXL.B #2,Rd		1	
	ROTXL.W Rd		1	
	ROTXL.W #2,Rd		1	
	ROTXL.L ERd		1	
	ROTXL.L #2,ERd		1	
ROTXR	ROTXR.B Rd		1	
	ROTXR.B #2,Rd		1	
	ROTXR.W Rd		1	
	ROTXR.W #2,Rd		1	
	ROTXR.L ERd		1	
	ROTXR.L #2,ERd		1	
RTE	RTE		2	2/3*1
RTS	RTS	Normal	2	1
		Advanced	2	2
SHAL	SHAL.B Rd		1	
	SHAL.B #2,Rd		1	
	SHAL.W Rd		1	
	SHAL.W #2,Rd		1	
	SHAL.L ERd		1	
	SHAL.L #2,ERd		1	
SHAR	SHAR.B Rd		1	
	SHAR.B #2,Rd		1	
	SHAR.W Rd		1	
	SHAR.W #2,Rd		1	
	SHAR.L ERd		1	
	SHAR.L #2,ERd		1	

	SHLL.L #2,ERd	1			
SHLR	SHLR.B Rd	1			
	SHLR.B #2,Rd	1			
	SHLR.W Rd	1			
	SHLR.W #2,Rd	1			
	SHLR.L ERd	1			
	SHLR.L #2,ERd	1			
SLEEP	SLEEP	1			
STC	STC.B CCR,Rd	1			
	STC.B EXR,Rd	1			
	STC.W CCR,@ERd	2			1
	STC.W EXR,@ERd	2			1
	STC.W CCR,@(d:16,ERd)	3			1
	STC.W EXR,@(d:16,ERd)	3			1
	STC.W CCR,@(d:32,ERd)	5			1
	STC.W EXR,@(d:32,ERd)	5			1
	STC.W CCR,@-ERd	2			1
	STC.W EXR,@-ERd	2			1
	STC.W CCR,@aa:16	3			1
	STC.W EXR,@aa:16	3			1
	STC.W CCR,@aa:32	4			1
	STC.W EXR,@aa:32	4			1
STM*4	STM.L (ERn-ERn+1),@-SP	2	4		
	STM.L (ERn-ERn+2),@-SP	2	6		
	STM.L (ERn-ERn+3),@-SP	2	8		
SUB	SUB.B Rs,Rd	1			
	SUB.W #xx:16,Rd	2			
	SUB.W Rs,Rd	1			
	SUB.L #xx:32,ERd	3			
	SUB.L ERs,ERd	1			

	Advanced	2	2	2/3 [1]
XOR	XOR.B #xx:8,Rd	1		
	XOR.B Rs,Rd	1		
	XOR.W #xx:16,Rd	2		
	XOR.W Rs,Rd	1		
	XOR.L #xx:32,ERd	3		
	XOR.L ERs,ERd	2		
XORC	XORC #xx:8,CCR	1		
	XORC #xx:8,EXR	2		

Notes: 1. 2 when EXR is invalid, 3 when valid.
2. When n bytes of data are transferred.
3. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS
4. Only registers ER0 to ER6 should be used when using the STM/LDM instru

Instruction	1	2	3	4	5	6	7
JMP@aa:24	R:W 2nd	Internal operation, 2 state	R:W EA				



Legend:

R:B	Byte-size read
R:W	Word-size read
W:B	Byte-size write
W:W	Word-size write
:M	Transfer of the bus is not performed immediately after this cycle
2nd	Address of 2nd word (3rd and 4th bytes)
3rd	Address of 3rd word (5th and 6th bytes)
4th	Address of 4th word (7th and 8th bytes)
5th	Address of 5th word (9th and 10th bytes)
NEXT	Start address of instruction following executing instruction
EA	Effective address
VEC	Vector address

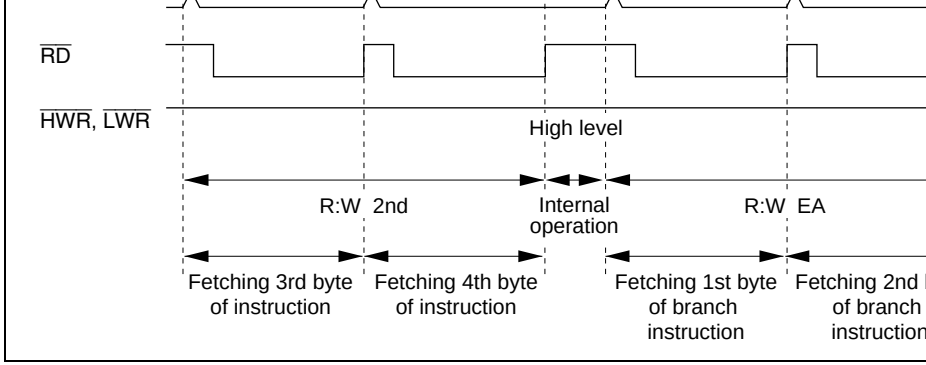


Figure A.1 Address Bus, $\overline{RD}$, $\overline{HWR}$, and $\overline{LWR}$ Timing (8-Bit Bus, Three-State Access, No Wait States)

ADDS #1/2/4,ERd	R:W NEXT							
ADDX #xx:8,Rd	R:W NEXT							
ADDX Rs,Rd	R:W NEXT							
AND.B #xx:8,Rd	R:W NEXT							
AND.B Rs,Rd	R:W NEXT							
AND.W #xx:16,Rd	R:W 2nd	R:W NEXT						
AND.W Rs,Rd	R:W NEXT							
AND.L #xx:32,ERd	R:W 2nd	R:W 3rd	R:W NEXT					
AND.L ERs,ERd	R:W 2nd	R:W NEXT						
ANDC #xx:8,CCR	R:W NEXT							
ANDC #xx:8,EXR	R:W 2nd	R:W NEXT						
BAND #xx:3,Rd	R:W NEXT							
BAND #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BAND #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BAND #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BAND #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BRA d:8 (BT d:8)	R:W NEXT	R:W EA						
BRN d:8 (BF d:8)	R:W NEXT	R:W EA						
BHI d:8	R:W NEXT	R:W EA						
BLS d:8	R:W NEXT	R:W EA						
BCC d:8 (BHS d:8)	R:W NEXT	R:W EA						
BCS d:8 (BLO d:8)	R:W NEXT	R:W EA						
BNE d:8	R:W NEXT	R:W EA						
BEQ d:8	R:W NEXT	R:W EA						
BVC d:8	R:W NEXT	R:W EA						
BVS d:8	R:W NEXT	R:W EA						
BPL d:8	R:W NEXT	R:W EA						

BRN d:16 (BF d:16)	R:W 2nd	Internal operation, 1 state	R:W EA					
BHI d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BLS d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BCC d:16 (BHS d:16)	R:W 2nd	Internal operation, 1 state	R:W EA					
BCS d:16 (BLO d:16)	R:W 2nd	Internal operation, 1 state	R:W EA					
BNE d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BEQ d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BVC d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BVS d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BPL d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BMI d:16	R:W 2nd	Internal operation, 1 state	R:W EA					
BGE d:16	R:W 2nd	Internal operation, 1 state	R:W EA					

BCLR #xx:3,Rd	R:W NEXT							
BCLR #xx:3,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR #xx:3,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR#xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BCLR#xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BCLR Rn,Rd	R:W NEXT							
BCLR Rn,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR Rn,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR Rn,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BCLR Rn,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BIAND #xx:3,Rd	R:W NEXT							
BIAND #xx:3, @ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BIAND #xx:3, @aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BIAND #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BIAND #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BILD #xx:3,Rd	R:W NEXT							
BILD #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BILD #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BILD #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B: EA	R:W:M NEXT				

BIOR #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BIST #xx:3,Rd	R:W NEXT							
BIST #xx:3,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BIST #xx:3,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BIST #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BIST #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BIXOR #xx:3,Rd	R:W NEXT							
BIXOR #xx:3, @ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BIXOR #xx:3, @aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BIXOR #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BIXOR #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BLD #xx:3,Rd	R:W NEXT							
BLD #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BLD #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BLD #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BLD #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BNOT #xx:3,Rd	R:W NEXT							
BNOT #xx:3,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				

BNOT Rn,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BNOT Rn,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BNOT Rn,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BOR #xx:3,Rd	R:W NEXT							
BOR #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BOR #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BOR #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BOR #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W NEXT			
BSET #xx:3,Rd	R:W NEXT							
BSET #xx:3,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BSET #xx:3,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BSET #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BSET #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BSET Rn,Rd	R:W NEXT							
BSET Rn,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BSET Rn,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA				
BSET Rn,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BSET Rn,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		

			NEXT					
BST #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA			
BST #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA		
BTST #xx:3,Rd	R:W NEXT							
BTST #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BTST #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BTST #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BTST #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BTST Rn,Rd	R:W NEXT							
BTST Rn,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BTST Rn,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BTST Rn,@aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BTST Rn,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
BXOR #xx:3,Rd	R:W NEXT							
BXOR #xx:3,@ERd	R:W 2nd	R:B EA	R:W:M NEXT					
BXOR #xx:3,@aa:8	R:W 2nd	R:B EA	R:W:M NEXT					
BXOR #xx:3, @aa:16	R:W 2nd	R:W 3rd	R:B EA	R:W:M NEXT				
BXOR #xx:3, @aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B EA	R:W:M NEXT			
CLRMAC	Cannot be used in this LSI							

DAS Rd		R:W NEXT							
DEC.B Rd		R:W NEXT							
DEC.W #1/2,Rd		R:W NEXT							
DEC.L #1/2,ERd		R:W NEXT							
DIVXS.B Rs,Rd		R:W 2nd	R:W NEXT	Internal operation, 11 states					
DIVXS.W Rs,ERd		R:W 2nd	R:W NEXT	Internal operation, 19 states					
DIVXU.B Rs,Rd		R:W NEXT	Internal operation, 11 states						
DIVXU.W Rs,ERd		R:W NEXT	Internal operation, 19 states						
EEPMOV.B		R:W 2nd	R:B EAs*1	R:B EAd*1	R:B EAs*2	W:B EAd*2	R:W NEXT		
EEPMOV.W		R:W 2nd	R:B EAs*1	R:B EAd*1	R:B EAs*2	W:B EAd*2	R:W NEXT		
EXTS.W Rd		R:W NEXT			← Repeated n times*2 →				
EXTS.L ERd		R:W NEXT							
EXTU.W Rd		R:W NEXT							
EXTU.L ERd		R:W NEXT							
INC.B Rd		R:W NEXT							
INC.W #1/2,Rd		R:W NEXT							
INC.L #1/2,ERd		R:W NEXT							
JMP @ERn		R:W NEXT	R:W EA						
JMP @aa:24		R:W 2nd	Internal operation, 1 state	R:W EA					
JMP @@aa:8	Advanced	R:W NEXT	R:W:M aa:8	R:W aa:8	Internal operation, 1 state	R:W EA			
JSR @ERn	Advanced	R:W NEXT	R:W EA	W:W:M Stack (H)	W:W Stack (L)				
JSR @aa:24	Advanced	R:W 2nd	Internal operation, 1 state	R:W EA	W:W:M Stack (H)	W:W Stack (L)			
JSR @@aa:8	Advanced	R:W NEXT	R:W:M aa:8	R:W aa:8	W:W:M Stack (H)	W:W Stack (L)	R:W EA		

LDC@(d:16,ERs), EXR	R:W 2nd	R:W 3rd	R:W NEXT	R:W EA				
LDC@(d:32,ERs), CCR	R:W 2nd	R:W 3rd	R:W 4th	R:W 5th	R:W NEXT	R:W EA		
LDC@(d:32,ERs), EXR	R:W 2nd	R:W 3rd	R:W 4th	R:W 5th	R:W NEXT	R:W EA		
LDC @ERs+,CCR	R:W 2nd	R:W NEXT	Internal operation, 1 state	R:W EA				
LDC @ERs+,EXR	R:W 2nd	R:W NEXT	Internal operation, 1 state	R:W EA				
LDC @aa:16,CCR	R:W 2nd	R:W 3rd	R:W NEXT	R:W EA				
LDC @aa:16,EXR	R:W 2nd	R:W 3rd	R:W NEXT	R:W EA				
LDC @aa:32,CCR	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	R:W EA			
LDC @aa:32,EXR	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	R:W EA			
LDM.L @SP+, (ERn-ERn+1)*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	R:W:M Stack (H)*3	R:W Stack (L)*3			
LDM.L @SP+, (ERn-ERn+2)*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	R:W:M Stack (H)*3	R:W Stack (L)*3			
LDM.L @SP+, (ERn-ERn+3)*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	R:W:M Stack (H)*3	R:W Stack (L)*3			
LDMAC ERs,MACH	Cannot be used in this LSI							
LDMAC ERs,MACL								
MAC @ERn+, @ERm+								
MOV.B #xx:8,Rd	R:W NEXT							
MOV.B Rs,Rd	R:W NEXT							
MOV.B @ERs,Rd	R:W NEXT	R:B EA						
MOV.B @(d:16,ERs),Rd	R:W 2nd	R:W NEXT	R:B EA					

MOV.B Rs,@ERd	R:W NEXT	W:B EA						
MOV.B Rs, @(d:16,ERd)	R:W 2nd	R:W NEXT	W:B EA					
MOV.B Rs, @(d:32,ERd)	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	W:B EA			
MOV.B Rs,@-ERd	R:W NEXT	Internal operation, 1 state	W:B EA					
MOV.B Rs,@aa:8	R:W NEXT	W:B EA						
MOV.B Rs,@aa:16	R:W 2nd	R:W NEXT	W:B EA					
MOV.B Rs,@aa:32	R:W 2nd	R:W 3rd	R:W NEXT	W:B EA				
MOV.W #xx:16,Rd	R:W 2nd	R:W NEXT						
MOV.W Rs,Rd	R:W NEXT							
MOV.W @ERs,Rd	R:W NEXT	R:W EA						
MOV.W @(d:16,ERs),Rd	R:W 2nd	R:W NEXT	R:W EA					
MOV.W @(d:32,ERs),Rd	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	R:W EA			
MOV.W @ERs+,Rd	R:W NEXT	Internal operation, 1 state	R:W EA					
MOV.W @aa:16,Rd	R:W 2nd	R:W NEXT	R:W EA					
MOV.W @aa:32,Rd	R:W 2nd	R:W 3rd	R:W NEXT	R:B EA				
MOV.W Rs,@ERd	R:W NEXT	W:W EA						
MOV.W Rs, @(d:16,ERd)	R:W 2nd	R:W NEXT	W:W EA					
MOV.W Rs, @(d:32,ERd)	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	W:W EA			
MOV.W Rs,@-ERd	R:W NEXT	Internal operation, 1 state	W:W EA					
MOV.W Rs,@aa:16	R:W 2nd	R:W NEXT	W:W EA					
MOV.W Rs,@aa:32	R:W 2nd	R:W 3rd	R:W NEXT	W:W EA				

MOV.L @ERs+, ERd	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	R:W:M EA	R:W EA+2			
MOV.L @aa:16, ERd	R:W 2nd	R:W:M 3rd	R:W NEXT	R:W:M EA	R:W EA+2			
MOV.L @aa:32, ERd	R:W 2nd	R:W:M 3rd	R:W 4th	R:W NEXT	R:W:M EA	R:W EA+2		
MOV.L ERs,@ERd	R:W 2nd	R:W:M NEXT	W:W:M EA	W:W EA+2				
MOV.L ERs, @(d:16,ERd)	R:W 2nd	R:W:M 3rd	R:W NEXT	W:W:M EA	W:W EA+2			
MOV.L ERs, @(d:32,ERd)	R:W 2nd	R:W:M 3rd	R:W:M 4th	R:W 5th	R:W NEXT	W:W:M EA	W:W EA+2	
MOV.L ERs,@-ERd	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	W:W:M EA	W:W EA+2			
MOV.L ERs, @aa:16	R:W 2nd	R:W:M 3rd	R:W NEXT	W:W:M EA	W:W EA+2			
MOV.L ERs, @aa:32	R:W 2nd	R:W:M 3rd	R:W 4th	R:W NEXT	W:W:M EA	W:W EA+2		
MOVFPE @aa:16,Rd	Cannot be used in this LSI							
MOVTPE Rs,@aa:16								
MULXS.B Rs,Rd	R:W 2nd	R:W NEXT	Internal operation, 11 states					
MULXS.W Rs,ERd	R:W 2nd	R:W NEXT	Internal operation, 19 states					
MULXU.B Rs,Rd	R:W NEXT	Internal operation, 11 states						
MULXU.W Rs,ERd	R:W NEXT	Internal operation, 19 states						
NEG.B Rd	R:W NEXT							
NEG.W Rd	R:W NEXT							
NEG.L ERd	R:W NEXT							
NOP	R:W NEXT							
NOT.B Rd	R:W NEXT							
NOT.W Rd	R:W NEXT							

ORC #xx:8,CCR	R:W NEXT							
ORC #xx:8,EXR	R:W 2nd	R:W NEXT						
POP.W Rn	R:W NEXT	Internal operation, 1 state	R:W EA					
POP.L ERn	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	R:W:M EA	R:W EA+2			
PUSH.W Rn	R:W NEXT	Internal operation, 1 state	W:W EA					
PUSH.L ERn	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	W:W:M EA	W:W EA+2			
ROTL.B Rd	R:W NEXT							
ROTL.B #2,Rd	R:W NEXT							
ROTL.W Rd	R:W NEXT							
ROTL.W #2,Rd	R:W NEXT							
ROTL.L ERd	R:W NEXT							
ROTL.L #2,ERd	R:W NEXT							
ROTR.B Rd	R:W NEXT							
ROTR.B #2,Rd	R:W NEXT							
ROTR.W Rd	R:W NEXT							
ROTR.W #2,Rd	R:W NEXT							
ROTR.L ERd	R:W NEXT							
ROTR.L #2,ERd	R:W NEXT							
ROTXL.B Rd	R:W NEXT							
ROTXL.B #2,Rd	R:W NEXT							
ROTXL.W Rd	R:W NEXT							
ROTXL.W #2,Rd	R:W NEXT							
ROTXL.L ERd	R:W NEXT							

RTE		R:W NEXT	R:W Stack (EXR)	R:W Stack (H)	R:W Stack (L)	Internal operation, 1 state	R:W		
RTS	Advanced	R:W NEXT	R:W:M Stack (H)	R:W Stack (L)	Internal operation, 1 state	R:W*4			
SHAL.B Rd		R:W NEXT							
SHAL.B #2,Rd		R:W NEXT							
SHAL.W Rd		R:W NEXT							
SHAL.W #2,Rd		R:W NEXT							
SHAL.L ERd		R:W NEXT							
SHAL.L #2,ERd		R:W NEXT							
SHAR.B Rd		R:W NEXT							
SHAR.B #2,Rd		R:W NEXT							
SHAR.W Rd		R:W NEXT							
SHAR.W #2,Rd		R:W NEXT							
SHAR.L ERd		R:W NEXT							
SHAR.L #2,ERd		R:W NEXT							
SHLL.B Rd		R:W NEXT							
SHLL.B #2,Rd		R:W NEXT							
SHLL.W Rd		R:W NEXT							
SHLL.W #2,Rd		R:W NEXT							
SHLL.L ERd		R:W NEXT							
SHLL.L #2,ERd		R:W NEXT							
SHLR.B Rd		R:W NEXT							
SHLR.B #2,Rd		R:W NEXT							
SHLR.W Rd		R:W NEXT							
SHLR.W #2,Rd		R:W NEXT							
SHLR.L ERd		R:W NEXT							
SHLR.L #2,ERd		R:W NEXT							

@(d:16,ERd)								
STC EXR, @(d:16,ERd)	R:W 2nd	R:W 3rd	R:W NEXT	W:W EA				
STC CCR, @(d:32,ERd)	R:W 2nd	R:W 3rd	R:W 4th	R:W 5th	R:W NEXT	W:W EA		
STC EXR, @(d:32,ERd)	R:W 2nd	R:W 3rd	R:W 4th	R:W 5th	R:W NEXT	W:W EA		
STC CCR,@-ERd	R:W 2nd	R:W NEXT	Internal operation, 1 state	W:W EA				
STC EXR,@-ERd	R:W 2nd	R:W NEXT	Internal operation, 1 state	W:W EA				
STC CCR,@aa:16	R:W 2nd	R:W 3rd	R:W NEXT	W:W EA				
STC EXR,@aa:16	R:W 2nd	R:W 3rd	R:W NEXT	W:W EA				
STC CCR,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	W:W EA			
STC EXR,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:W NEXT	W:W EA			
STM.L (ERn-ERn+1), @-SP*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	W:W:M Stack (H) *3	W:W Stack (L) *3			
STM.L (ERn-ERn+2), @-SP*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	W:W:M Stack (H) *3	W:W Stack (L) *3			
STM.L (ERn-ERn+3), @-SP*9	R:W 2nd	R:W:M NEXT	Internal operation, 1 state	W:W:M Stack (H) *3	W:W Stack (L) *3			
STMAC MACH,ERd	Cannot be used in this LSI							
STMAC MACL,ERd								
SUB.B Rs,Rd	R:W NEXT							
SUB.W #xx:16,Rd	R:W 2nd	R:W NEXT						
SUB.W Rs,Rd	R:W NEXT							
SUB.L #xx:32,ERd	R:W 2nd	R:W 3rd	R:W NEXT					
SUB.L ERs,ERd	R:W NEXT							

XOR.B Rs,Rd		R:W NEXT							
XOR.W #xx:16,Rd		R:W 2nd	R:W NEXT						
XOR.W Rs,Rd		R:W NEXT							
XOR.L #xx:32,ERd		R:W 2nd	R:W 3rd	R:W NEXT					
XOR.L ERs,ERd		R:W 2nd	R:W NEXT						
XORC #xx:8,CCR		R:W NEXT							
XORC #xx:8,EXR		R:W 2nd	R:W NEXT						
Reset exception handling	Advanced	R:W:M VEC	R:W VEC+2	Internal operation, 1 state	R:W*5				
Interrupt exception handling	Advanced	R:W*6	Internal operation, 1 state	W:W Stack (L)	W:W Stack (H)	W:W Stack (EXR)	R:W:M VEC	R:W VEC+2	Interna operat 1 state

Notes: 1. EAs is the contents of ER5. EAd is the contents of ER6.

2. EAs is the contents of ER5. EAd is the contents of ER6. Both registers are by 1 after execution of the instruction. n is the initial value of R4L or R4. If n bus cycles are not executed.
3. Repeated two times to save or restore two registers, three times for three r four times for four registers.
4. Start address after return.
5. Start address of the program.
6. Prefetch address, equal to two plus the PC value pushed onto the stack. In from sleep mode or software standby mode the read operation is replaced internal operation.
7. Start address of the interrupt-handling routine.
8. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS
9. Only registers ER0 to ER6 should be used when using the STM/LDM instru

H'EFFF										
	MRB	CHNE	DISEL	—	—	—	—	—	—	
	DAR									
	CRA									
	CRB									
H'FE80	HICR2	—	—	—	—	—	IBFIE4	IBFIE3	—	HIF
H'FE84	IDR3	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0	
H'FE85	ODR3	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0	
H'FE86	STR3	DBU	DBU	DBU	DBU	C/$\overline{D}$	DBU	IBF	OBF	
H'FE8C	IDR4	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0	
H'FE8D	ODR4	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0	
H'FE8E	STR4	DBU	DBU	DBU	DBU	C/$\overline{D}$	DBU	IBF	OBF	
H'FED8	KBCRH0	KBIOE	KCLKI	KDI	KBFSEL	KBIE	KBF	PER	KBS	Key buf con
H'FED9	KBCRL0	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0	
H'FEDA	KBBR0	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0	
H'FEDC	KBCRH1	KBIOE	KCLKI	KDI	KBFSEL	KBIE	KBF	PER	KBS	
H'FEDD	KBCRL1	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0	
H'FEDE	KBBR1	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0	
H'FEE0	KBCRH2	KBIOE	KCLKI	KDI	KBFSEL	KBIE	KBF	PER	KBS	
H'FEE1	KBCRL2	KBE	KCLKO	KDO	—	RXCR3	RXCR2	RXCR1	RXCR0	
H'FEE2	KBBR2	KB7	KB6	KB5	KB4	KB3	KB2	KB1	KB0	
H'FEE4	KBCOMP	IrE	IrCKS2	IrCKS1	IrCKS0	KBADE	KBCH2	KBCH1	KBCH0	IrDA exp A/D
H'FEE6	DDCSWR	SWE	SW	IE	IF	CLR3	CLR2	CLR1	CLR0	IIC(

H'FEEF	DTCERB	DTCEB7	DTCEB6	DTCEB5	DTCEB4	DTCEB3	DTCEB2	DTCEB1	DTCEB0	
H'FEF0	DTCERC	DTCEC7	DTCEC6	DTCEC5	DTCEC4	DTCEC3	DTCEC2	DTCEC1	DTCEC0	
H'FEF1	DTCERD	DTCED7	DTCED6	DTCED5	DTCED4	DTCED3	DTCED2	DTCED1	DTCED0	
H'FEF2	DTCERE	DTCEE7	DTCEE6	DTCEE5	DTCEE4	DTCEE3	DTCEE2	DTCEE1	DTCEE0	
H'FEF3	DTVECR	SWDTE	DTVEC6	DTVEC5	DTVEC4	DTVEC3	DTVEC2	DTVEC1	DTVEC0	
H'FEF4	ABRKCR	CMF	—	—	—	—	—	—	BIE	In
H'FEF5	BARA	A23	A22	A21	A20	A19	A18	A17	A16	co
H'FEF6	BARB	A15	A14	A13	A12	A11	A10	A9	A8	
H'FEF7	BARC	A7	A6	A5	A4	A3	A2	A1	—	
H'FF80	FLMCR1	FWE	SWE	—	—	EV	PV	E	P	FI
H'FF81	FLMCR2	FLER	—	—	—	—	—	ESU	PSU	
H'FF82	PCSR	—	—	—	—	—	PWCKB	PWCKA	—	P'
	EBR1	—	—	—	—	—	—	EB9/—	EB8/—	FI
H'FF83	SYSCR2	KWUL1	KWUL0	P6PUE	—	SDE	CS4E	CS3E	HI12E	H
	EBR2	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0	FI
H'FF84	SBYCR	SSBY	STS2	STS1	STS0	—	SCK2	SCK1	SCK0	S
H'FF85	LPWRCR	DTON	LSON	NESEL	EXCLE	—	—	—	—	
H'FF86	MSTPCRH	MSTP15	MSTP14	MSTP13	MSTP12	MSTP11	MSTP10	MSTP9	MSTP8	
H'FF87	MSTPCRL	MSTP7	MSTP6	MSTP5	MSTP4	MSTP3	MSTP2	MSTP1	MSTP0	
H'FF88	SMR1	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	S
	ICCR1	ICE	IEIC	MST	TRS	ACKE	BBSY	IRIC	SCP	II
H'FF89	BRR1									S
	ICSR1	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB	II
H'FF8A	SCR1	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	S
H'FF8B	TDR1									
H'FF8C	SSR1	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
H'FF8D	RDR1									

H'FF92	FRCH									
H'FF93	FRCL									
H'FF94	OCRAH									
	OCRBH									
H'FF95	OCRAL									
	OCRBL									
H'FF96	TCR	IEDGA	IEDGB	IEDGC	IEDGD	BUFEA	BUFEB	CKS1	CKS0	
H'FF97	TOCR	ICRDMS	OCRAMS	ICRS	OCRS	OEA	OEB	OLVLA	OLVLB	
H'FF98	ICRAH									
	OCRARH									
H'FF99	ICRAL									
	OCRARL									
H'FF9A	ICRBH									
	OCRAFH									
H'FF9B	ICRBL									
	OCRAFL									
H'FF9C	ICRCH									
	OCRDMH	0	0	0	0	0	0	0	0	
H'FF9D	ICRCL									
	OCRDML									
H'FF9E	ICRDH									
H'FF9F	ICRDL									
H'FFA0	SMR2	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	SC
	DADRAH	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	PW
	DACR	TEST	PWME	—	—	OEB	OEA	OS	CKS	
H'FFA1	BRR2									SC
	DADRAL	DA5	DA4	DA3	DA2	DA1	DA0	CFS	—	PW

H'FFA7	DADRBL	DA5	DA4	DA3	DA2	DA1	DA0	CFS	REGS	
	DACNTL							—	REGS	
H'FFA8	TCSR0	OVF	WT/$\overline{IT}$	TME	RSTS	RST/$\overline{NMI}$	CKS2	CKS1	CKS0	W
	TCNT0 (write)									
H'FFA9	TCNT0 (read)									
H'FFAA	PAODR	PA7ODR	PA6ODR	PA5ODR	PA4ODR	PA3ODR	PA2ODR	PA1ODR	PA0ODR	P
H'FFAB	PAPIN (read)	PA7PIN	PA6PIN	PA5PIN	PA4PIN	PA3PIN	PA2PIN	PA1PIN	PA0PIN	
	PADDR (write)	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	PA0DDR	
H'FFAC	P1PCR	P17PCR	P16PCR	P15PCR	P14PCR	P13PCR	P12PCR	P11PCR	P10PCR	
H'FFAD	P2PCR	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR	P20PCR	
H'FFAE	P3PCR	P37PCR	P36PCR	P35PCR	P34PCR	P33PCR	P32PCR	P31PCR	P30PCR	
H'FFB0	P1DDR	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR	
H'FFB1	P2DDR	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR	
H'FFB2	P1DR	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR	P10DR	
H'FFB3	P2DR	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR	P20DR	
H'FFB4	P3DDR	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	
H'FFB5	P4DDR	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR	
H'FFB6	P3DR	P37DR	P36DR	P35DR	P34DR	P33DR	P32DR	P31DR	P30DR	
H'FFB7	P4DR	P47DR	P46DR	P45DR	P44DR	P43DR	P42DR	P41DR	P40DR	
H'FFB8	P5DDR	—	—	—	—	—	P52DDR	P51DDR	P50DDR	
H'FFB9	P6DDR	P67DDR	P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR	
H'FFBA	P5DR	—	—	—	—	—	P52DR	P51DR	P50DR	
H'FFBB	P6DR	P67DR	P66DR	P65DR	P64DR	P63DR	P62DR	P61DR	P60DR	
H'FFBC	PBODR	PB7ODR	PB6ODR	PB5ODR	PB4ODR	PB3ODR	PB2ODR	PB1ODR	PB0ODR	
H'FFBD	PBPIN (read)	PB7PIN	PB6PIN	PB5PIN	PB4PIN	PB3PIN	PB2PIN	PB1PIN	PB0PIN	
	P8DDR (write)	—	P86DDR	P85DDR	P84DDR	P83DDR	P82DDR	P81DDR	P80DDR	

H'FFC2	IER	IRQ7E	IRQ6E	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	Inte con
H'FFC3	STCR	IICS	IICX1	IICX0	IICE	FLSHE	—	ICKS1	ICKS0	Sys
H'FFC4	SYSCR	CS2E	IOSE	INTM1	INTM0	XRST	NMIEG	HIE	RAME	
H'FFC5	MDCR	EXPE	—	—	—	—	—	MDS1	MDS0	
H'FFC6	BCR	ICIS1	ICIS0	BRSTRM	BRSTS1	BRSTS0	—	IOS1	IOS0	Bus con
H'FFC7	WSCR	RAMS	RAM0	ABW	AST	WMS1	WMS0	WC1	WC0	
H'FFC8	TCR0	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	TM
H'FFC9	TCR1	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	TM
H'FFCA	TCSR0	CMFB	CMFA	OVF	ADTE	OS3	OS2	OS1	OS0	
H'FFCB	TCSR1	CMFB	CMFA	OVF	—	OS3	OS2	OS1	OS0	
H'FFCC	TCORA0									
H'FFCD	TCORA1									
H'FFCE	TCORB0									
H'FFCF	TCORB1									
H'FFD0	TCNT0									
H'FFD1	TCNT1									
H'FFD2	PWOERB	OE15	OE14	OE13	OE12	OE11	OE10	OE9	OE8	PW
H'FFD3	PWOERA	OE7	OE6	OE5	OE4	OE3	OE2	OE1	OE0	
H'FFD4	PWDPRB	OS15	OS14	OS13	OS12	OS11	OS10	OS9	OS8	
H'FFD5	PWDPRA	OS7	OS6	OS5	OS4	OS3	OS2	OS1	OS0	
H'FFD6	PWSL	PWCKE	PWCKS	—	—	RS3	RS2	RS1	RS0	
H'FFD7	PWDR0 to PWDR15									
H'FFD8	SMR0	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	SC
	ICCR0	ICE	IEIC	MST	TRS	ACKE	BBSY	IRIC	SCP	IIC
H'FFD9	BRR0									SC
	ICSR0	ESTP	STOP	IRTR	AASX	AL	AAS	ADZ	ACKB	IIC

H'FFDF	ICMR0	MLS	WAIT	CKS2	CKS1	CKS0	BC2	BC1	BC0	
	SAR0	SVA6	SVA5	SVA4	SVA3	SVA2	SVA1	SVA0	FS	
H'FFE0	ADDRAH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A
H'FFE1	ADDRAL	AD1	AD0	—	—	—	—	—	—	
H'FFE2	ADDRBH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE3	ADDRBL	AD1	AD0	—	—	—	—	—	—	
H'FFE4	ADDRCH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE5	ADDRCL	AD1	AD0	—	—	—	—	—	—	
H'FFE6	ADDRDH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE7	ADDRDL	AD1	AD0	—	—	—	—	—	—	
H'FFE8	ADCSR	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
H'FFE9	ADCR	TRGS1	TRGS0	—	—	—	—	—	—	
H'FFEA	TCSR1	OVF	WT/$\overline{\text{IT}}$	TME	PSS	RST/$\overline{\text{NMI}}$	CKS2	CKS1	CKS0	W
	TCNT1 (write)									
H'FFEB	TCNT1 (read)									
H'FFF0	HICR	—	—	—	—	—	IBFIE2	IBFIE1	FGA20E	H
	TCRX	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	TI
	TCRY	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	TI
H'FFF1	KMIMR	KMIMR7	KMIMR6	KMIMR5	KMIMR4	KMIMR3	KMIMR2	KMIMR1	KMIMR0	In co
	TCSRX	CMFB	CMFA	OVF	ICF	OS3	OS2	OS1	OS0	TI
	TCSRY	CMFB	CMFA	OVF	ICIE	OS3	OS2	OS1	OS0	TI
H'FFF2	KMPCR	KM7PCR	KM6PCR	KM5PCR	KM4PCR	KM3PCR	KM2PCR	KM1PCR	KM0PCR	P
	TICRR									TI
	TCORAY									TI
H'FFF3	KMIMRA	KMIMR15	KMIMR14	KMIMR13	KMIMR12	KMIMR11	KMIMR10	KMIMR9	KMIMR8	In co
	TICRF									TI
	TCORBY									TI

	TCORAX									TM
H'FFF7	TCORBX									
H'FFF8	DADR0									D/A
H'FFF9	DADR1									
H'FFFA	DACR	DAOE1	DAOE0	DAE	—	—	—	—	—	
H'FFFC	IDR2	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0	HIF
	TCONRI	SIMOD1	SIMOD0	SCONE	ICST	HFINV	VFINV	HIINV	VIINV	Tim con
H'FFFD	ODR2	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0	HIF
	TCONRO	HOE	VOE	CLOE	CBOE	HOINV	VOINV	CLOINV	CBOINV	Tim con
H'FFFE	STR2	DBU	DBU	DBU	DBU	C/$\overline{D}$	DBU	IBF	OBF	HIF
	TCONRS	TMRX/Y	ISGENE	HOMOD1	HOMOD0	VOMOD1	VOMOD0	CLMOD1	CLMOD0	Tim
H'FFFF	SEDGR	VEDG	HEDG	CEDG	HFEDG	VFEDG	PREQF	IHI	IVI	con

H'FE80	HICR2	MSTP2 = 0	MSTP2 = 0	—
H'FE84	IDR3			
H'FE85	ODR3			
H'FE86	STR3			
H'FE8C	IDR4			
H'FE8D	ODR4			
H'FE8E	STR4			
H'FED8	KBCRH0	MSTP2 = 0	MSTP2 = 0	—
H'FED9	KBCRL0			
H'FEDA	KBBR0			
H'FEDC	KBCRH1			
H'FEDD	KBCRL1			
H'FEDE	KBBR1			
H'FEE0	KBCRH2			
H'FEE1	KBCRL2			
H'FEE2	KBBR2			
H'FEE4	KBCOMP	No conditions	No conditions	No conditions
H'FEE6	DDCSWR	MSTP4 = 0	MSTP4 = 0	—
H'FEE8	ICRA	No conditions	No conditions	No conditions
H'FEE9	ICRB			
H'FEEA	ICRC			
H'FEEB	ISR			
H'FEEC	ISCRH			
H'FEED	ISCRL			
H'FEEE	DTCERA	No conditions	—	—
H'FEEF	DTCERB			
H'FEF0	DTCERC			
H'FEF1	DTCERD			
H'FEF2	DTCERE			

	EBR1	FLSHE = 1 in STCR			FLSHE = 1 in STCR		FLSHE = 1 in STCR
H'FF83	SYSCR2	FLSHE = 0 in STCR			FLSHE = 0 in STCR		—
	EBR2	FLSHE = 1 in STCR			FLSHE = 1 in STCR		FLSHE = 1 in STCR
H'FF84	SBYCR	FLSHE = 0 in STCR			FLSHE = 0 in STCR		FLSHE = 0 in STCR
H'FF85	LPWRCR						
H'FF86	MSTPCRH						
H'FF87	MSTPCRL						
H'FF88	SMR1	MSTP6 = 0, IICE = 0 in STCR			MSTP6 = 0, IICE = 0 in STCR		MSTP6 = 0, IICE = 0 in STCR
	ICCR1	MSTP3 = 0, IICE = 1 in STCR			MSTP3 = 0, IICE = 1 in STCR		—
H'FF89	BRR1	MSTP6 = 0, IICE = 0 in STCR			MSTP6 = 0, IICE = 0 in STCR		MSTP6 = 0, IICE = 0 in STCR
	ICSR1	MSTP3 = 0, IICE = 1 in STCR			MSTP3 = 0, IICE = 1 in STCR		—
H'FF8A	SCR1	MSTP6 = 0			MSTP6 = 0		MSTP6 = 0
H'FF8B	TDR1						
H'FF8C	SSR1						
H'FF8D	RDR1						
H'FF8E	SCMR1	MSTP6 = 0, IICE = 0 in STCR			MSTP6 = 0, IICE = 0 in STCR		MSTP6 = 0, IICE = 0 in STCR
	ICDR1	MSTP3 = 0, IICE = 1 in STCR	ICE = 1 in ICCR1		MSTP3 = 0, IICE = 1 in STCR	ICE = 1 in ICCR1	—
	SARX1		ICE = 0 in ICCR1			ICE = 0 in ICCR1	
H'FF8F	ICMR1		ICE = 1 in ICCR1			ICE = 1 in ICCR1	
	SAR1		ICE = 0 in ICCR1			ICE = 0 in ICCR1	
H'FF90	TIER	MSTP13 = 0			MSTP13 = 0		MSTP13 = 0
H'FF91	TCSR						
H'FF92	FRCH						
H'FF93	FRCL						

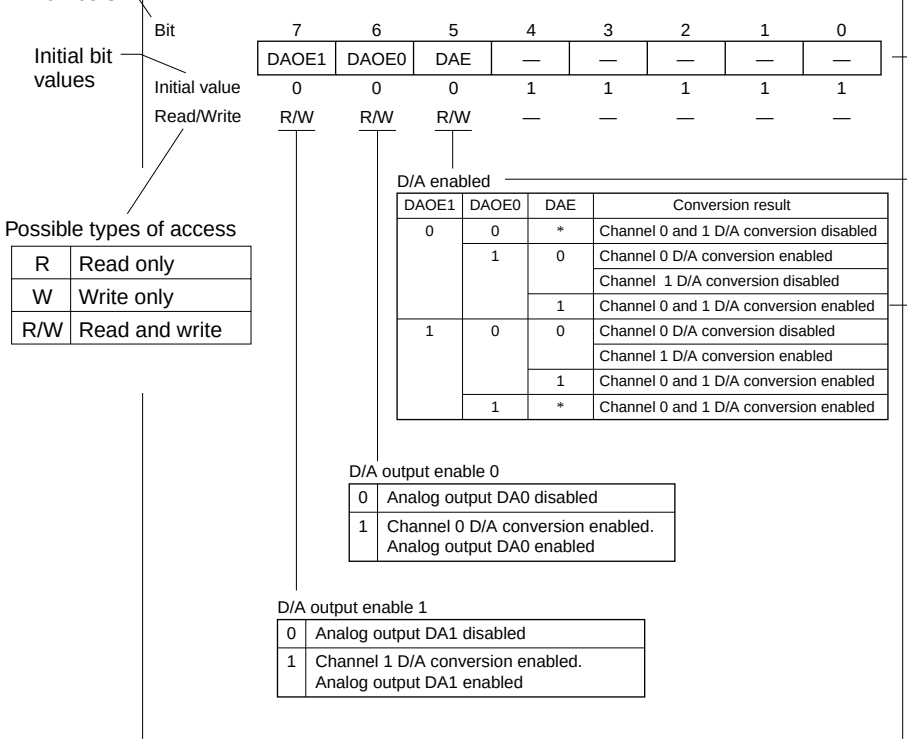
H'FF97	TOCR										
H'FF98	ICRAH		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRARH		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF99	ICRAL		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRARL		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF9A	ICRBH		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRAFH		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF9B	ICRBL		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRAFL		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF9C	ICRCH		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRDMH		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF9D	ICRCL		ICRS = 0 in TOCR			ICRS = 0 in TOCR				ICRS = TOCR	
	OCRDML		ICRS = 1 in TOCR			ICRS = 1 in TOCR				ICRS = TOCR	
H'FF9E	ICRDH										
H'FF9F	ICRDL										
H'FFA0	SMR2	MSTP5 = 0, IICE = 0 in STCR			MSTP5 = 0, IICE = 0 in STCR			MSTP5 = 0, IICE = 0 in STCR			
	DADRAH	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/ DADRB		MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/ DADRB		MSTP11 = 0, IICE = 1 in STCR	REGS = in DACN DADRB		
	DACR		REGS = 1 in DACNT/ DADRB			REGS = 1 in DACNT/ DADRB			REGS = in DACN DADRB		

	DADRBH	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/ DADRB	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT/ DADRB	MSTP11 = 0, IICE = 1 in STCR	REGS = 0 in DACNT DADRB
	DACNTH		REGS = 1 in DACNT/ DADRB		REGS = 1 in DACNT/ DADRB		REGS = 1 in DACNT DADRB
H'FFA7	DADRBL		REGS = 0 in DACNT/ DADRB		REGS = 0 in DACNT/ DADRB		REGS = 0 in DACNT DADRB
	DACNTL		REGS = 1 in DACNT/ DADRB		REGS = 1 in DACNT/ DADRB		REGS = 1 in DACNT DADRB
H'FFA8	TCSR0	No conditions		No conditions		No conditions	
	TCNT0 (write)						
H'FFA9	TCNT0 (read)						
H'FFAA	PAODR0	No conditions		No conditions		No conditions	
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'FFB6	P3DR						
H'FFB7	P4DR						

H'FFBE	P7PIN (read)			
	PBDDR (write)			
H'FFBF	P8DR			
H'FFC0	P9DDR			
H'FFC1	P9DR			
H'FFC2	IER	No conditions	No conditions	No conditions
H'FFC3	STCR	No conditions	No conditions	No conditions
H'FFC4	SYSCR			
H'FFC5	MDCR			
H'FFC6	BCR			
H'FFC7	WSCR			
H'FFC8	TCR0	MSTP12 = 0	MSTP12 = 0	MSTP12 = 0
H'FFC9	TCR1			
H'FFCA	TCSR0			
H'FFCB	TCSR1			
H'FFCC	TCORA0			
H'FFCD	TCORA1			
H'FFCE	TCORB0			
H'FFCF	TCORB1			
H'FFD0	TCNT0			
H'FFD1	TCNT1			
H'FFD2	PWOERB	No conditions	No conditions	—
H'FFD3	PWOERA			
H'FFD4	PWDPRB			
H'FFD5	PWDPRA			

H'FFDB	TDR0					
H'FFDC	SSR0					
H'FFDD	RDR0					
H'FFDE	SCMR0	MSTP7 = 0, IICE = 0 in STCR		MSTP7 = 0, IICE = 0 in STCR		MSTP7 = 0, IICE = 0 in STCR
	ICDR0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0	MSTP4 = 0, IICE = 1 in STCR	ICE = 1 in ICCR0	—
	SARX0		ICE = 0 in ICCR0		ICE = 0 in ICCR0	
H'FFDF	ICMR0		ICE = 1 in ICCR0		ICE = 1 in ICCR0	
	SAR0		ICE = 0 in ICCR0		ICE = 0 in ICCR0	
H'FFE0	ADDRAH	MSTP9 = 0		MSTP9 = 0		MSTP9 = 0
H'FFE1	ADDRAL					
H'FFE2	ADDRBH					
H'FFE3	ADDRBL					
H'FFE4	ADDRCH					
H'FFE5	ADDRCL					
H'FFE6	ADDRDH					
H'FFE7	ADDRDL					
H'FFE8	ADCSR					
H'FFE9	ADCR					
H'FFEA	TCSR1	No conditions		No conditions		No conditions
	TCNT1 (write)					
H'FFEB	TCNT1 (read)					
H'FFF0	HICR	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR		—
	TCRX	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	—		—
	TCRY		TMRX/Y = 1 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR		MSTP8 = 0, HIE = 0 in SYSCR

	TCORAY		TMRX/Y = 1 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR
H'FFF3	KMIMRA	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	MSTP2 = 0, HIE = 1 in SYSCR
	TICRF	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	—	—
	TCORBY		TMRX/Y = 1 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR
H'FFF4	IDR1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	—
	TCNTX	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	—	
	TCNTY		TMRX/Y = 1 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR
H'FFF5	ODR1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	—
	TCORC	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	—	
	TISR		TMRX/Y = 1 in TCONRS	MSTP8 = 0, HIE = 0 in SYSCR	MSTP8 = 0, HIE = 0 in SYSCR
H'FFF6	STR1	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	—
	TCORAX	MSTP8 = 0, HIE = 0 in SYSCR	TMRX/Y = 0 in TCONRS	—	
H'FFF7	TCORBX				
H'FFF8	DADR0	MSTP10 = 0		MSTP10 = 0	MSTP10 = 0
H'FFF9	DADR1				
H'FFFA	DACR				
H'FFFC	IDR2	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	—
	TCONRI	MSTP8 = 0, HIE = 0 in SYSCR		—	
H'FFFD	ODR2	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCONRO	MSTP8 = 0, HIE = 0 in SYSCR		—	
H'FFFE	STR2	MSTP2 = 0, HIE = 1 in SYSCR		MSTP2 = 0, HIE = 1 in SYSCR	
	TCONRS	MSTP8 = 0, HIE = 0 in SYSCR		—	
H'FFFF	SEDGR				



DAOE1	DAOE0	DAE	Conversion result
0	0	*	Channel 0 and 1 D/A conversion disabled
	1	0	Channel 0 D/A conversion enabled
			Channel 1 D/A conversion disabled
		1	Channel 0 and 1 D/A conversion enabled
1	0	0	Channel 0 D/A conversion disabled
			Channel 1 D/A conversion enabled
		1	Channel 0 and 1 D/A conversion enabled
	1	*	Channel 0 and 1 D/A conversion enabled

DTC data tra

0	Byte-siz
1	Word-si

DTC transfer mode

0	Destination side area or block ar
1	Source side is r or block area

DTC mode

0	0	Normal mode
	1	Repeat mode
1	0	Block transfer r
	1	—

Destination address mode

0	—	DAR is fixed
1	0	DAR is incremented after a transfer (by 1 when Sz = 0; by 2 when Sz =
	1	DAR is decremented after a transfe (by 1 when Sz = 0; by 2 when Sz =

Source address mode

0	—	SAR is fixed
1	0	SAR is incremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)
	1	SAR is decremented after a transfer (by 1 when Sz = 0; by 2 when Sz = 1)

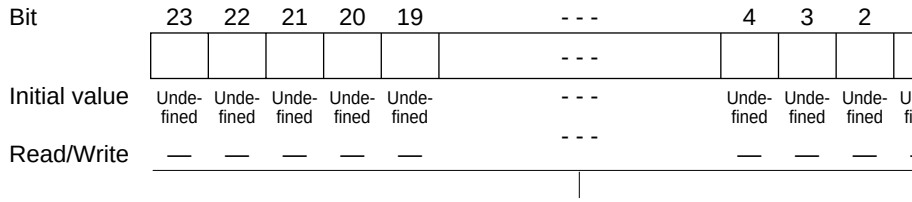
DTC interrupt select

0	After a data transfer ends, the CPU interrupt is disabled unless the transfer counter is 0
1	After a data transfer ends, the CPU interrupt is enabled

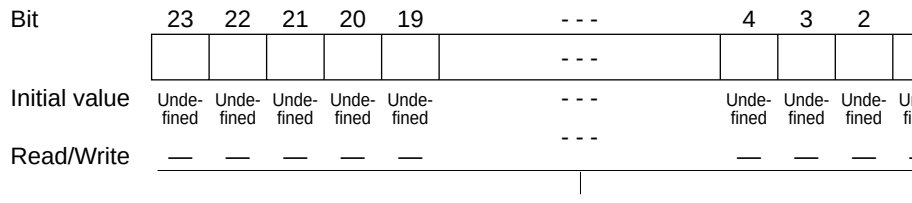
DTC chain transfer enable

0	End of DTC data transfer
1	DTC chain transfer

SAR—DTC Source Address Register **H'EC00–H'EFFF**



DAR—DTC Destination Address Register **H'EC00–H'EFFF**



Specifies the number of DTC data transfers

CRB—DTC Transfer Count Register B **H'EC00–H'EFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Unde-fined	Und fine
Read/Write	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Specifies the number of DTC block data transfers

HICR2—Host Interface Control Register 2 **H'FE80**

Bit	7	6	5	4	3	2	1
	—	—	—	—	—	IBFIE4	IBFIE3
Initial value	1	1	1	1	1	0	0
Slave R/W	—	—	—	—	—	R/W	R/W
Host R/W	—	—	—	—	—	—	—

Input data register full interrupt enable

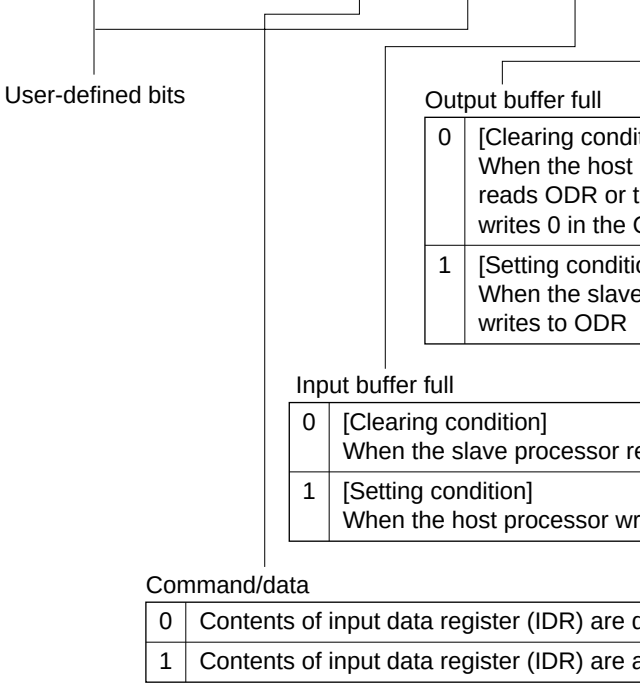
IBFIE4	IBFIE3	Description
—	0	Input data register (IDR3) receive complete interrupt is d
—	1	Input data register (IDR3) receive complete interrupt is e
0	—	Input data register (IDR4) receive complete interrupt is d
1	—	Input data register (IDR4) receive complete interrupt is e

Stores host data bus contents at rise of $\overline{IOW}$ when $\overline{CS}$ is low

ODR3—Output Data Register 3 **H'FE85**
ODR4—Output Data Register 4 **H'FE8D**

Bit	7	6	5	4	3	2	1
	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1
Initial value	—	—	—	—	—	—	—
Slave R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Host R/W	R	R	R	R	R	R	R

ODR contents are output to the host data bus when HA0 is low, $\overline{CS}$ is low, and $\overline{IOR}$ is low


User-defined bits
Output buffer full
0
[Clearing condi
When the host
reads ODR or t
writes 0 in the C
1
[Setting conditio
When the slave
writes to ODR
Input buffer full
0
[Clearing condition]
When the slave processor re
1
[Setting condition]
When the host processor wr
Command/data
0
Contents of input data register (IDR) are d
1
Contents of input data register (IDR) are a

Keyboard stop

0	0 stop bit re
1	1 stop bit re

Parity error

0	[Clearing condition] Read PER when PER = then write 0 in PER
1	[Setting condition] When an odd parity err

Keyboard buffer register full

0	[Clearing condition] Read KBF when KBF =1, then write 0 i
1	[Setting conditions] • When data has been received norma KBFSEL = 1, and has been transferre KBBR (keyboard buffer register full fl • When a KCLK falling edge has been while KBFSEL = 0 (KCLK interrupt fl

Keyboard interrupt enable

0	Interrupt requests are disabled
1	Interrupt requests are enabled

Keyboard buffer register full select

0	KBF bit is used as KCLK fall interrupt flag
1	KBF bit is used as keyboard buffer full flag

Keyboard data in

0	KD I/O pin is low
1	KD I/O pin is high

Keyboard clock in

0	KCLK I/O pin is low
1	KCLK I/O pin is high

Keyboard in/out enable

0	The keyboard buffer controller is non-operational (KCLK and KD signal pins have port functions)
1	The keyboard buffer controller is enabled for transmission and reception (KCLK and KD signal pins are in the bus drive state)

Note: * Only 0 can be written, to clear the flag.

Receive counter

RXCR3	RXCR2	RXCR1	RXCR0	Receive c
0	0	0	0	
			1	St
		1	0	I
			1	I
	1	0	0	I
			1	I
		1	0	I
			1	I
1	0	0	0	I
			1	I
		1	0	Pa
			1	
	1	—	—	

Keyboard data out

0	Keyboard buffer controller data I/O pin is low
1	Keyboard buffer controller data I/O pin is high

Keyboard clock out

0	Keyboard buffer controller clock I/O pin is low
1	Keyboard buffer controller clock I/O pin is high

Keyboard enable

0	Loading of receive data into KBBR is disabled
1	Loading of receive data into KBBR is enabled

Stores receive data

Keyboard comparator control

Bit 3	Bit 2	Bit 1	Bit 0	A/D converter channel 6 input	A/D char
KBADE	KBCH2	KBCH1	KBCH0		
0	—	—	—	AN6	
1	0	0	0	CIN0	
			1	CIN1	
		1	0	CIN2	
			1	CIN3	
	1	0	0	CIN4	
			1	CIN5	
		1	0	CIN6	
			1	CIN7	

IrDA Clock select 2 to 0

0	0	0	B × 3/16 (3/16 of the bit rate)
		1	ϕ/2
	1	0	ϕ/4
		1	ϕ/8
1	0	0	ϕ/16
		1	ϕ/32
	1	0	ϕ/64
		1	ϕ/128

IrDA enable

0	The TxD2/IrTxD and RxD2/IrRxD pins function as TxD2 and RxD2
1	The TxD2/IrTxD and RxD2/IrRxD pins function as IrTxD and IrRxD

Bit 3	Bit 2	Bit 1	Bit 0	Description
CLR3	CLR2	CLR1	CLR0	
0	0	—	—	Setting prohibited
	1	0	0	Setting prohibited
			1	IIC0 internal latch cleared
		1	0	IIC1 internal latch cleared
			1	IIC0 and IIC1 internal latches
1	—	—	—	Invalid setting

DDC mode switch interrupt flag

0	No interrupt is requested when automatic format switching is executed [Clearing condition] When 0 is written in IF after reading IF = 1
1	An interrupt is requested when automatic format switching is executed [Setting condition] When a falling edge is detected on the SCL pin when SWE = 1

DDC mode switch interrupt enable bit

0	Interrupt when automatic format switching is executed is disable
1	Interrupt when automatic format switching is executed is enable

DDC mode switch

0	IIC channel 0 is used with the I^2C bus format [Clearing conditions] • When 0 is written by software • When a falling edge is detected on the SCL pin when SWE = 1
1	IIC channel 0 is used in formatless mode [Setting condition] When 1 is written in SW after reading SW = 0

DDC mode switch enable

0	Automatic switching of IIC channel 0 from formatless mode to I^2C bus format is dis
1	Automatic switching of IIC channel 0 from formatless mode to I^2C bus format is ena

Notes: 1. Only 0 can be written, to clear the flag.
2. Always read as 1.

Interrupt control level

0	Corresponding interrupt source is control level 0 (non
1	Corresponding interrupt source is control level 1 (prio

Correspondence between Interrupt Sources and ICR Settings

Register	Bits						
	7	6	5	4	3	2	1
ICRA	IRQ0	IRQ1	IRQ2 IRQ3	IRQ4 IRQ5	IRQ6 IRQ7	DTC	Watchdog timer 0
ICRB	A/D converter	Free-running timer	—	—	8-bit timer channel 0	8-bit timer channel 1	8-bit timer channels X, Y
ICRC	SCI channel 0	SCI channel 1	SCI channel 2	IIC channel 0 (option)	IIC channel 1 (option)	—	—

0	[Clearing conditions] • Cleared by reading IRQnF when set to 1, then writing 0 in IRQnF • When interrupt exception handling is executed while low-level de is set (IRQnSCB = IRQnSCA = 0) and $\overline{\text{IRQn}}$ input is high* • When IRQn interrupt exception handling is executed while falling or both-edge detection is set (IRQnSCB = 1 or IRQnSCA = 1)*
1	[Setting conditions] • When $\overline{\text{IRQn}}$ input goes low when low-level detection is set (IRQnSCB = IRQnSCA = 0) • When a falling edge occurs in $\overline{\text{IRQn}}$ input while falling edge detec set (IRQnSCB = 0, IRQnSCA = 1) • When a rising edge occurs in $\overline{\text{IRQn}}$ input while rising edge detect set (IRQnSCB = 1, IRQnSCA = 0) • When a falling or rising edge occurs in $\overline{\text{IRQn}}$ input while both-edg detection is set (IRQnSCB = IRQnSCA = 1)

Notes: n = 7 to 0

* When a product, in which a DTC is incorporated, is used in th following settings, the corresponding flag bit is not automatica cleared even when exception handling, which is a clear condi executed and the bit is held at 1.

(1) When DTCEA3 is set to 1 (ADI is set to an interrupt sourc IRQ4F flag is not automatically cleared.

(2) When DTCEA2 is set to 1 (ICIA is set to an interrupt sour IRQ5F flag is not automatically cleared.

(3) When DTCEA1 is set to 1 (ICIB is set to an interrupt sour IRQ6F flag is not automatically cleared.

(4) When DTCEA0 is set to 1 (OCIA is set to an interrupt sou IRQ7F flag is not automatically cleared.

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

Note: * Only 0 can be written, to clear the flag.

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IRQ7 to IRQ4 sense control A and B

ISCRL

Bit	7	6	5	4	3	2	1
	IRQ3SCB	IRQ3SCA	IRQ2SCB	IRQ2SCA	IRQ1SCB	IRQ1SCA	IRQ0SCB
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IRQ3 to IRQ0 sense control A and B

ISCRH bits 7 to 0 ISCRL bits 7 to 0		Description
IRQ7SCB to IRQ0SCB	IRQ7SCA to IRQ0SCA	
0	0	Interrupt request generated at $\overline{\text{IRQ7}}$ to input at low level
	1	Interrupt request generated at falling e of $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ input
1	0	Interrupt request generated at rising ed of $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ input
	1	Interrupt request generated at both fall and rising edges of $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ inpu

0	DTC activation by interrupt is disabled [Clearing conditions] • When data transfer ends with the DISEL bit set to 1 • When the specified number of transfers end
1	DTC activation by interrupt is enabled [Holding condition] When the DISEL bit is 0 and the specified number of transfers have not ended

DTVECR—DTC Vector Register H'FEF3

Bit	7	6	5	4	3	2	1	
	SWDTE	DTVEC6	DTVEC5	DTVEC4	DTVEC3	DTVEC2	DTVEC1	D
Initial value	0	0	0	0	0	0	0	
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W	

Sets vector number for DTC software a

DTC software activation enable

0	DTC software activation is disabled [Clearing condition] When the DISEL bit is 0 and the specified number of transfers h not ended
1	DTC software activation is enabled [Holding conditions] • When data transfer ends with the DISEL bit set to 1 • When the specified number of transfers end • During software-activated deta transfer

Note: * A value of 1 can always be written to the SWDTE bit, but 0 can only be writt after 1 is read.

0	Address break di
1	Address break en

Condition match flag

0	[Clearing condition] When address break interrupt exception handling is executed
1	[Setting condition] When address set by BARA to BARC is prefetched while BIE =

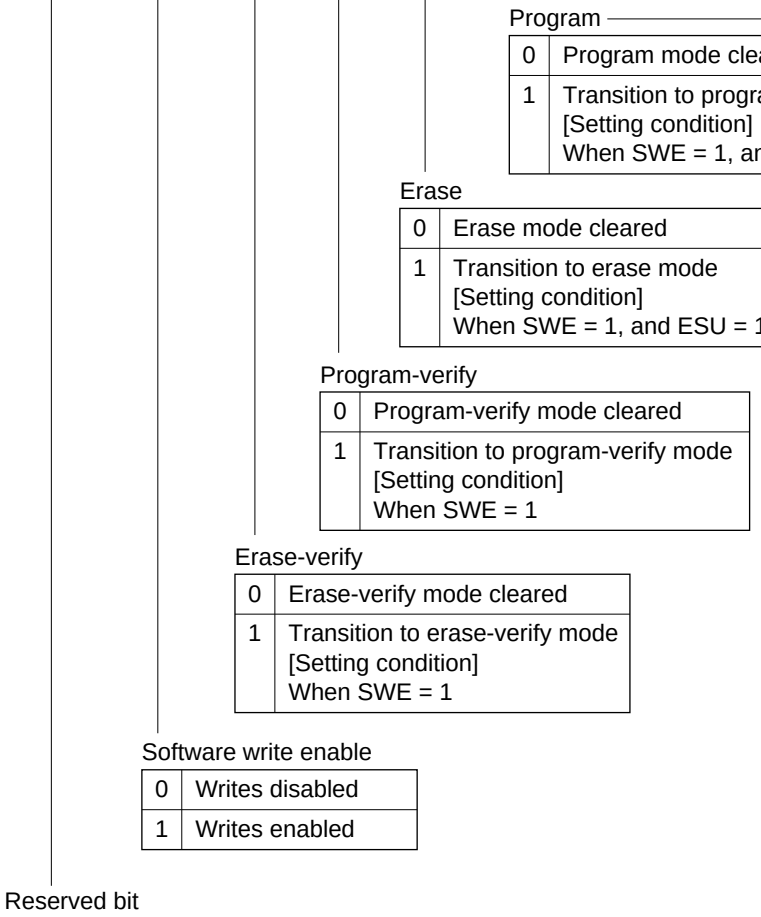
Specifies address (bits 23 to 16) at which address break is to be

Bit	7	6	5	4	3	2	1	
BARB	A15	A14	A13	A12	A11	A10	A9	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Specifies address (bits 15 to 8) at which address break is to be g

Bit	7	6	5	4	3	2	1	
BARC	A7	A6	A5	A4	A3	A2	A1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Specifies address (bits 7 to 1) at which address break is to be ge


Program
0
Program mode cle
1
Transition to progra
[Setting condition]
When SWE = 1, ar
Erase
0
Erase mode cleared
1
Transition to erase mode
[Setting condition]
When SWE = 1, and ESU = 1
Program-verify
0
Program-verify mode cleared
1
Transition to program-verify mode
[Setting condition]
When SWE = 1
Erase-verify
0
Erase-verify mode cleared
1
Transition to erase-verify mode
[Setting condition]
When SWE = 1
Software write enable
0
Writes disabled
1
Writes enabled
Reserved bit

Program setup

0	Program setup
1	Program setup [Setting condit When SWE =

Erase setup

0	Erase setup cleared
1	Erase setup [Setting condition] When SWE = 1

Flash memory error

0	Flash memory is operating normally Flash memory program/erase protection (error protection) is disab [Clearing condition] Reset or hardware standby mode
1	An error has occurred during flash memory programming/erasing Flash memory program/erase protection (error protection) is enab [Setting condition] See section 22.8.3 or 23.8.3, Error Protection

PWSL		PCSR		Description
Bit 7	Bit 6	Bit 2	Bit 1	
PWCKE	PWCKS	PWCKB	PWCKA	
0	—	—	—	Clock input is disabled
1	0	—	—	ϕ (system clock) is selec
	1	0	0	ϕ/2 is selected
			1	ϕ/4 is selected
		1	0	ϕ/8 is selected
			1	ϕ/16 is selected

Host interface enable

0	Host interface fu are disabled
1	Host interface fu are enabled

CS3 enable

0	Host interface pin ch functions disabled
1	Host interface pin ch functions enabled

CS4 enable

0	Host interface pin chann functions disabled
1	Host interface pin chann functions enabled

Shutdown enable

0	Host interface pin shutdown function d
1	Host interface pin shutdown function e

Port 6 input pull-up extra

0	Standard current specification is selected for port 6 input pull-up function
1	Current-limit specification is selected for port 6 MO pull-up function

Key wakeup level 1 and 0

0	0	Standard input level is selected as port 6 input level
	1	Input level 1 is selected as port 6 input level
1	0	Input level 2 is selected as port 6 input level
	1	Input level 3 is selected as port 6 input level

Bit	7	6	5	4	3	2	1	
EBR2	EB7	EB6	EB5	EB4	EB3	EB2	EB1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	

Notes: 1. In normal mode, these bits cannot be modified and are always read as 0.
2. Bits EB8 and EB9 are not present in the 64-kbyte versions; they must not

Erase Blocks

Block (Size)		Addresses
128-kbyte versions	64-kbyte versions	
EB0 (1 kbyte)	EB0 (1 kbyte)	H'(00)0000 to H'(00)03FF
EB1 (1 kbyte)	EB1 (1 kbyte)	H'(00)0400 to H'(00)07FF
EB2 (1 kbyte)	EB2 (1 kbyte)	H'(00)0800 to H'(00)0BFF
EB3 (1 kbyte)	EB3 (1 kbyte)	H'(00)0C00 to H'(00)0FFF
EB4 (28 kbytes)	EB4 (28 kbytes)	H'(00)1000 to H'(00)7FFF
EB5 (16 kbytes)	EB5 (16 kbytes)	H'(00)8000 to H'(00)BFFF
EB6 (8 kbytes)	EB6 (8 kbytes)	H'(00)C000 to H'(00)DFFF
EB7 (8 kbytes)	EB7 (8 kbytes)	H'00E000 to H'00FFFF
EB8 (32 kbytes)	—	H'010000 to H'017FFF
EB9 (32 kbytes)	—	H'018000 to H'01FFFF

System clock select 2 to 0

0	0	0	Bus master is in high-spe
		1	Medium-speed clock = φ/
	1	0	Medium-speed clock = φ/
		1	Medium-speed clock = φ/
1	0	0	Medium-speed clock = φ/
		1	Medium-speed clock = φ/
	1	—	—

Standby timer select 2 to 0

0	0	0	Standby time = 8192 states
		1	Standby time = 16384 states
	1	0	Standby time = 32768 states
		1	Standby time = 65536 states
1	0	0	Standby time = 131072 states
		1	Standby time = 262144 states
	1	0	Reserved
		1	Standby time = 16 states*

Note: * This setting must not be used in the flash memory vers

Software standby

0	Transition to sleep mode after execution of SLEEP instruction in high-speed or medium-speed mode Transition to subsleep mode on execution of SLEEP instruction in subactive
1	Transition to software standby mode, subactive mode, or watch mode after e of SLEEP instruction in high-speed mode or medium-speed mode Transition to watch mode or high-speed mode after execution of SLEEP inst subactive mode

0	Subclock input from EXCL pin is
1	Subclock input from EXCL pin is

Noise elimination sampling frequency select

0	Sampling at ϕ divided by 32
1	Sampling at ϕ divided by 4

Low-speed on flag

0	• When a SLEEP instruction is executed in high-speed mode medium-speed mode, a transition is made to sleep mode, so standby mode, or watch mode* • When a SLEEP instruction is executed in subactive mode, a is made to watch mode, or directly to high-speed mode • After watch mode is cleared, a transition is made to high-spe
1	• When a SLEEP instruction is executed in high-speed mode transition is made to watch mode or subactive mode* • When a SLEEP instruction is executed in subactive mode, a is made to subsleep mode or watch mode • After watch mode is cleared, a transition is made to subactiv

Note: * When a transition is made to watch mode or subactive mo high-speed mode must be set.

Direct-transfer on flag

0	• When a SLEEP instruction is executed in high-speed mode or medium-speed a transition is made to sleep mode, software standby mode, or watch mode* • When a SLEEP instruction is executed in subactive mode, a transition is mad subsleep mode or watch mode
1	• When a SLEEP instruction is executed in high-speed mode or medium-speed a transition is made directly to subactive mode*, or a transition is made to sle or software standby mode • When a SLEEP instruction is executed in subactive mode, a transition is mad to high-speed mode, or a transition is made to subsleep mode

Note: * When a transition is made to watch mode or subactive mode, high-speed m must be set.

Read/Write R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W

Module stop

0	Module stop mode is cleared
1	Module stop mode is set

The correspondence between MSTPCR bits and on-chip supporting modules is sh

Register	Bit	Module
MSTPCRH	MSTP15	—
	MSTP14*	Data transfer controller (DTC)
	MSTP13	16-bit free-running timer (FRT)
	MSTP12	8-bit timers (TMR0, TMR1)
	MSTP11	8-bit PWM timer (PWM), 14-bit PWM timer (PWMX)
	MSTP10	D/A converter
	MSTP9	A/D converter
	MSTP8	8-bit timers (TMRX, TMRY), timer connection
MSTPCRL	MSTP7	Serial communication interface 0 (SCI0)
	MSTP6	Serial communication interface 1 (SCI1)
	MSTP5	Serial communication interface 2 (SCI2)
	MSTP4*	I^2C bus interface (IIC) channel 0 (option)
	MSTP3*	I^2C bus interface (IIC) channel 1 (option)
	MSTP2*	Host interface (HIF), keyboard buffer controller (PS2)
	MSTP1*	—
	MSTP0*	—

Notes: Do not set bit 15 to 1. Bits 1 and 0 can be read and written but do not affec

* Must be set to 1 in the H8S/2144 Group.

Clock sele

0	0	φ
	1	φ/
1	0	φ/
	1	φ/

Multiprocessor mode

0	Multiprocessor functio
1	Multiprocessor format

Stop bit length

0	1 stop bit
1	2 stop bits

Parity mode

0	Even parity
1	Odd parity

Parity enable

0	Parity bit addition and checking disabled
1	Parity bit addition and checking enabled

Character length

0	8-bit data
1	7-bit data*

Note: * When 7-bit data is selected, the MSB (bit 7) of TDR is not transmitted. Also, LSB-first/ MSB-first selection is not available.

Communication mode

0	Asynchronous mode
1	Synchronous mode

Start condition/stop condi
prohibit

0	Writing 0 issues a st stop condition, in co with the BBSY flag
1	Reading always retu value of 1; writing is

I²C bus interface interrupt req

0	Waiting for transfer, or tr progress
1	Interrupt requested

Note: For the clearing and se
conditions, see section
I²C Bus Control Regist

Bus busy

0	Bus is free [Clearing condition] When a stop condition is dete
1	Bus is busy [Setting condition] When a start condition is dete

I²C bus interface interrupt enable

0	Interrupts disabled
1	Interrupts enabled

Acknowledge bit judgement selection

0	The value of the acknowledge bit is and continuous transfer is performe
1	If the acknowledge bit is 1, continu transfer is interrupted

I²C bus interface enable

0	I²C bus interface module disabled, with SCL and SDA signal pins set to port function SAR and SARX can be accessed
1	I²C bus interface module enabled for transfer operations (pins SCL and SDA are driving the bus) ICMR and ICDR can be accessed

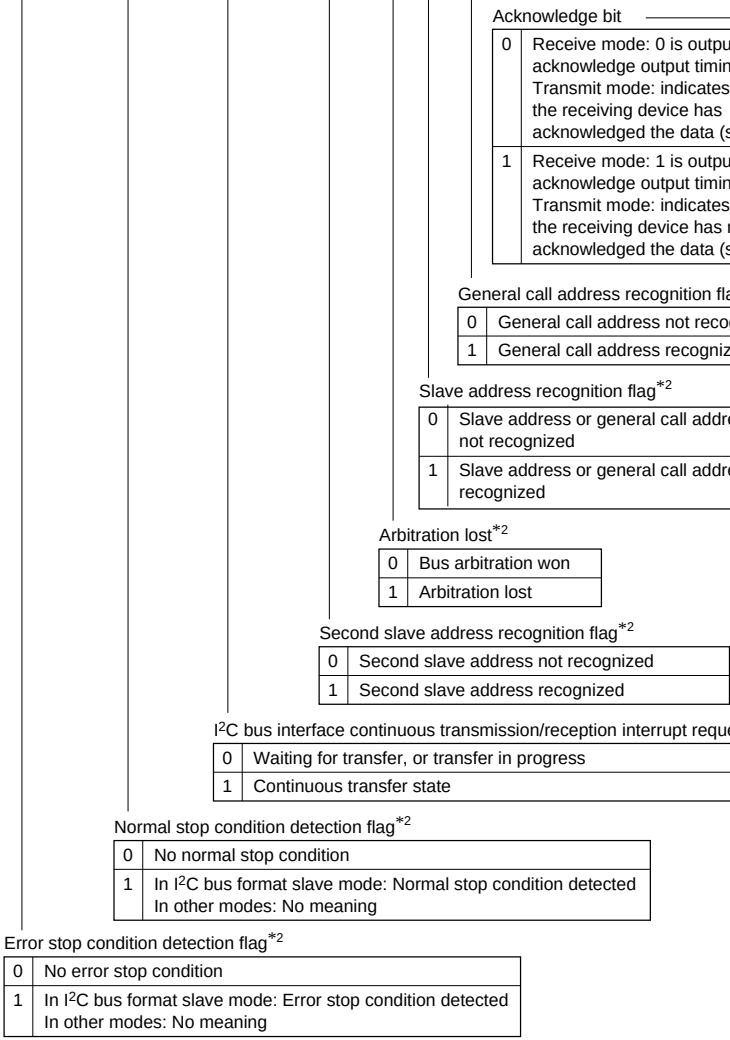
Master/slave select (MST), transmit/receive se

0	0	Slave receive mode
	1	Slave transmit mode
1	0	Master receive mode
	1	Master transmit mode

Note: For details, see section 16.2.5, I²C Bus
Register (ICCR).

Note: * Only 0 can be written, to clear the flag.

Sets the serial transmit/receive bit rate



Notes: 1. Only 0 can be written, to clear the flag.
2. For the clearing and setting conditions, see section 16.2.6, I²C Bus Status Register (ICSR).

Clock enable 1 and 0

0	0	Asynchronous mode	Internal clock/SCK functions as I/O po
		Synchronous mode	Internal clock/SCK functions as serial
	1	Asynchronous mode	Internal clock/SCK functions as clock
		Synchronous mode	Internal clock/SCK functions as serial
1	0	Asynchronous mode	External clock/SCK functions as clock i
		Synchronous mode	External clock/SCK functions as serial
	1	Asynchronous mode	External clock/SCK functions as clock i
		Synchronous mode	External clock/SCK functions as serial

Transmit end interrupt enable

0	Transmit-end interrupt (TEI) request disabled
1	Transmit-end interrupt (TEI) request enabled

Multiprocessor interrupt enable

0	Multiprocessor interrupts disabled (normal reception [Clearing conditions] • When the MPIE bit is cleared to 0 • When data with MPB = 1 is received
1	Multiprocessor interrupts enabled Receive interrupt (RXI) requests, receive-error inter requests, and setting of the RDRF, FER, and ORER SSR are disabled until data with the multiprocessor 1 is received

Transmit interrupt enable

0	Transmit-data-empty interrupt (TXI) request disabled
1	Transmit-data-empty interrupt (TXI) request enabled

Receive enable

0	Reception disabled
1	Reception enabled

Receive interrupt enable

0	Receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) request disabled
1	Receive-data-full interrupt (RXI) request and receive-error interrupt (ERI) request enabled

Transmit enable

0	Transmission disabled
1	Transmission enabled

Stores serial receive data

TDR1—Transmit Data Register 1 **H'FF8B**
TDR2—Transmit Data Register 2 **H'FFA3**
TDR0—Transmit Data Register 0 **H'FFDB**

Bit	7	6	5	4	3	2	1	
Initial value	1	1	1	1	1	1	1	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Stores serial transmit data

Multiprocessor bit trans

0	Data with a 0 mult bit is transmitted
1	Data with a 1 mult bit is transmitted

Multiprocessor bit

0	[Clearing condition] When data with a 0 mult bit is received
1	[Setting condition] When data with a 1 mult bit is received

Transmit end

0	[Clearing conditions] • When 0 is written in TDRE after reading • When the DTC is activated by a TXI inte writes data to TDR
1	[Setting conditions] • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the l a 1-byte serial transmit character

Parity error

0	[Clearing condition] When 0 is written in PER after reading PER = 1
1	[Setting condition] When, in reception, the number of 1 bits in the re data plus the parity bit does not match the parity (even or odd) specified by the O/$\overline{E}$ bit in SMR

Framing error

0	[Clearing condition] When 0 is written in FER after reading FER = 1
1	[Setting condition] When the SCI checks the stop bit at the end of the receive data when reception ends, and the stop bit is 0

Overrun error

0	[Clearing condition] When 0 is written in ORER after reading ORER = 1
1	[Setting condition] When the next serial reception is completed while RDRF = 1

Receive data register full

0	[Clearing conditions] • When 0 is written in RDRF after reading RDRF = 1 • When the DTC is activated by an RXI interrupt and reads data from RDR
1	[Setting condition] When serial reception ends normally and receive data is transferred from RSR to RD

Transmit data register empty

0	[Clearing conditions] • When 0 is written in TDRE after reading TDRE = 1 • When the DTC is activated by a TXI interrupt and writes data to TDR
1	[Setting conditions] • When the TE bit in SCR is 0 • When data is transferred from TDR to TSR and data can be written in TDR

Note: * Only 0 can be written, to

Serial commu
interface mod

0	Normal S
1	Setting p

Data invert

0	TDR contents are transmitted without mod Receive data is stored in RDR without mod
1	TDR contents are inverted before being tra Receive data is stored in RDR in inverted f

Data transfer direction

0	TDR contents are transmitted LSB-first Receive data is stored in RDR LSB-first
1	TDR contents are transmitted MSB-first Receive data is stored in RDR MSB-first

ICDRR

Bit	7	6	5	4	3	2	1
	ICDRR7	ICDRR6	ICDRR5	ICDRR4	ICDRR3	ICDRR2	ICDRR1
Initial value	—	—	—	—	—	—	—
Read/Write	R	R	R	R	R	R	R

ICDRS

Bit	7	6	5	4	3	2	1
	ICDRS7	ICDRS6	ICDRS5	ICDRS4	ICDRS3	ICDRS2	ICDRS1
Initial value	—	—	—	—	—	—	—
Read/Write	—	—	—	—	—	—	—

ICDRT

Bit	7	6	5	4	3	2	1
	ICDRT7	ICDRT6	ICDRT5	ICDRT4	ICDRT3	ICDRT2	ICDRT1
Initial value	—	—	—	—	—	—	—
Read/Write	W	W	W	W	W	W	W

TDRE, RDRF (internal flags)

Bit	—
	TDRE
Initial value	0
Read/Write	—

Note: For details, see section 16.2.1, I^2C Bus Data Register (ICDR).

Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Slave address

Format

SARX

Bit	7	6	5	4	3	2	1	
	SVAX6	SVAX5	SVAX4	SVAX3	SVAX2	SVAX1	SVAX0	F
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Second slave address

Format select

DDCSWR Bit 6	SAR Bit 0	SARX Bit 0	Operating Mode
SW	FS	FSX	
0	0	0	I²C bus format • SAR and SARX slave addresses recognized
		1	I²C bus format • SAR slave address recognized • SARX slave address ignored
	1	0	I²C bus format • SAR slave address ignored • SARX slave address recognized
		1	Synchronous serial format • SAR and SARX slave addresses ignored
1	0	0	Formatless mode (start/stop conditions not detected) • Acknowledge bit used
		1	
	1	0	
		1	Formatless mode* (start/stop conditions not detected) • No acknowledge bit

Note: * Do not set this mode when automatic switching to the I²C bus format is performed by means of the DDCSWR setting.

Bit counter

BC2	BC1	BC0	Synchronous serial format	I
0	0	0	8	
		1	1	
	1	0	2	
		1	3	
1	0	0	4	
		1	5	
	1	0	6	
		1	7	

Serial clock select

IICX	CKS2	CKS1	CKS0	Clock
0	0	0	0	ϕ/28
			1	ϕ/40
		1	0	ϕ/48
			1	ϕ/64
	1	0	0	ϕ/80
			1	ϕ/100
		1	0	ϕ/112
			1	ϕ/128
1	0	0	0	ϕ/56
			1	ϕ/80
		1	0	ϕ/96
			1	ϕ/128
	1	0	0	ϕ/160
			1	ϕ/200
		1	0	ϕ/224
			1	ϕ/256

Wait insertion bit

0	Data and acknowledge bits transferred consecutively
1	Wait inserted between data and acknowledge bits

MSB-first/LSB-first select*

0	MSB-first
1	LSB-first

Note: * Do not set this bit to 1 when the I^2C bus format is used.

0	Timer overflow in request (FOVI) is
1	Timer overflow in request (FOVI) is

Output compare interrupt B

0	Output compare inter request B (OCIB) is d
1	Output compare inter request B (OCIB) is e

Output compare interrupt A enable

0	Output compare interrupt requ (OCIA) is disabled
1	Output compare interrupt requ (OCIA) is enabled

Input capture interrupt D enable

0	Input capture interrupt request D (ICID) is
1	Input capture interrupt request D (ICID) is

Input capture interrupt C enable

0	Input capture interrupt request C (ICIC) is dis
1	Input capture interrupt request C (ICIC) is ena

Input capture interrupt B enable

0	Input capture interrupt request B (ICIB) is disabled
1	Input capture interrupt request B (ICIB) is enabled

Input capture interrupt A enable

0	Input capture interrupt request A (ICIA) is disabled
1	Input capture interrupt request A (ICIA) is enabled

	disabl
1	FRC i compa

Timer overflow flag

0	[Clearing condition Read OVF when C then write 0 in OVI
1	[Setting condition] When FRC change H'FFFF to H'0000

Output compare flag B

0	[Clearing condition] Read OCFB when OCFB = 1, then writ
1	[Setting condition] When FRC = OCRB

Output compare flag A

0	[Clearing condition] Read OCFA when OCFA = 1, then write 0
1	[Setting condition] When FRC = OCRA

Input capture flag D

0	[Clearing condition] Read ICFD when ICFD = 1, then write 0 in ICFD
1	[Setting condition] When an input capture signal is received

Input capture flag C

0	[Clearing condition] Read ICFC when ICFC = 1, then write 0 in ICFC
1	[Setting condition] When an input capture signal is received

Input capture flag B

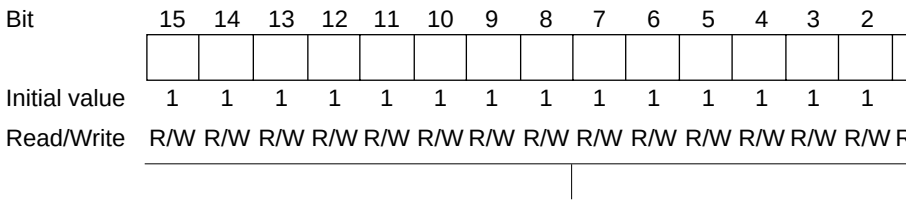
0	[Clearing condition] Read ICFB when ICFB = 1, then write 0 in ICFB
1	[Setting condition] When an input capture signal causes the FRC value to be transferred to ICRB

Input capture flag A

0	[Clearing condition] Read ICFA when ICFA = 1, then write 0 in ICFA
1	[Setting condition] When an input capture signal causes the FRC value to be transferred to ICRA

Note: * Only 0 can be written in bits 7 to 1, to clear the flags.

OCRA/OCRB—Output Compare Register A/B H'FF94



Clock select

0	0	φ/2 internal clock
	1	φ/8 internal clock
1	0	φ/32 internal cloc
	1	External clock so (rising edge)

Buffer enable B

0	ICRD is not used as a bu register for input capture
1	ICRD is used as a buffer register for input capture

Buffer enable A

0	ICRC is not used as a buffer regist input capture A
1	ICRC is used as a buffer register fo capture A

Input edge select D

0	Capture on the falling edge of FTID
1	Capture on the rising edge of FTID

Input edge select C

0	Capture on the falling edge of FTIC
1	Capture on the rising edge of FTIC

Input edge select B

0	Capture on the falling edge of FTIB
1	Capture on the rising edge of FTIB

Input edge select A

0	Capture on the falling edge of FTIA
1	Capture on the rising edge of FTIA

0	0 output at co match B
1	1 output at co match B

Output level A

0	0 output at compare match A
1	1 output at compare match A

Output enable B

0	Output compare B output d
1	Output compare B output e

Output enable A

0	Output compare A output disabl
1	Output compare A output enabl

Output compare register select

0	OCRA register selected
1	OCRB register selected

Input capture register select

0	ICRA, ICRB, and ICRC registers selected
1	OCRAR, OCRAF, and OCRDM registers selecte

Output compare A mode select

0	OCRA set to normal operating mode
1	OCRA set to operating mode using OCRAR and OCRAF

Input capture D mode select

0	ICRD set to normal operating mode
1	ICRD set to operating mode using OCRDM

Used for OCRA operation when OCRAMS = 1 in TOCR
(For details, see section 11.2.4, Output Compare Regist
AR and AF (OCRAR, OCRAF).)

OCRDM—Output Compare Register DM **H'FF9C**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Used for ICRD operation when ICRDMS = 1 in TOCR
(For details, see section 11.2.5, Output Compare Regist
DM (OCRDM).)

ICRA—Input Capture Register A **H'FF98**
ICRB—Input Capture Register B **H'FF9A**
ICRC—Input Capture Register C **H'FF9C**
ICRD—Input Capture Register D **H'FF9E**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Stores FRC value when input capture signal is input
(ICRC and ICRD can be used for buffer operation.
For details, see section 11.2.3, Input Capture Regist
A to D (ICRA to ICRD).)

DADRA	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	CF
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

DADRB	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	CF
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Register select (DADRB only)

0	DADRA and DADRB can be a
1	DACR and DACNT can be ac

Carrier frequency select

0	Base cycle = resolution (T) × 64 DADR is H'0401 to H'FFFD
1	Base cycle = resolution (T) × 256 DAD is H'0103 to H'FFFF

D/A conversion data

Clock select

0	Operates at resolution system clock cycle tim
1	Operates at resolution system clock cycle tim

Output select

0	Direct PWM output
1	Inverted PWM output

Output enable A

0	PWM (D/A) channel A output (PWX0 output pin) disabled
1	PWM (D/A) channel A output (PWX0 output pin) enabled

Output enable B

0	PWM (D/A) channel B output (PWX1 output pin) disabled
1	PWM (D/A) channel B output (PWX1 output pin) enabled

PWM enable

0	DACNT operates as a 14-bit up-counter
1	DACNT halts at H'0003

Test mode

0	PWM (D/A) in user state: normal operation
1	PWM (D/A) in test state: correct conversion results unobtainable

Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Register select

0	DADRA and DADRB can be a
1	DACR and DACNT can be ac

Up-counter

0	0	0	φ/2
		1	φ/64
	1	0	φ/128
		1	φ/512
1	0	0	φ/204
		1	φ/819
	1	0	φ/327
		1	φ/131

Reset or NMI

0	NMI interrupt requested
1	Internal reset requested

Reserved bit

Timer enable

0	TCNT is initialized to H'00 and halted
1	TCNT counts

Timer mode select

0	Interval timer mode: Sends the CPU an interval timer interr request (WOVI) when TCNT overflows
1	Watchdog timer mode: Generates a reset or NMI interrupt TCNT overflows

Overflow flag

0	[Clearing conditions] • Write 0 in the TME bit • Read TCSR when OVF = 1*, then write 0 in OVF
1	[Setting condition] When TCNT overflows (changes from H'FF to H'00) (When internal reset request generation is selected in watchdog timer mode, OVF is cleared automatically by the internal reset.)

Note: * When OVF is polled and the interval timer interrupt is disabled, OVF = 1 must be read at least twice.

Note: * Only 0 can be written, to clear the flag.

Up-counter

PAODR—Port A Output Data Register **H'FFAA**

Bit	7	6	5	4	3	2	1	
	PA7ODR	PA6ODR	PA5ODR	PA4ODR	PA3ODR	PA2ODR	PA1ODR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Output data for port A pins

PAPIN—Port A Input Data Register **H'FFAB (R)**

Bit	7	6	5	4	3	2	1	
	PA7PIN	PA6PIN	PA5PIN	PA4PIN	PA3PIN	PA2PIN	PA1PIN	
Initial value	—*	—*	—*	—*	—*	—*	—*	
Read/Write	R	R	R	R	R	R	R	

Port A pin states

Note: * Determined by state of pins PA7 to PA0.

PADDR—Port A Data Direction Register **H'FFAB (W)**

Bit	7	6	5	4	3	2	1	
	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	P
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

Specification of input or output for port A pins

P2PCR—Port 2 MOS Pull-Up Control Register H'FFAD

Bit	7	6	5	4	3	2	1
	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Control of port 2 built-in MOS input pull-ups

P3PCR—Port 3 MOS Pull-Up Control Register H'FFAE

Bit	7	6	5	4	3	2	1
	P37PCR	P36PCR	P35PCR	P34PCR	P33PCR	P32PCR	P31PCR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Control of port 3 built-in MOS input pull-ups

P1DDR—Port 1 Data Direction Register H'FFB0

Bit	7	6	5	4	3	2	1
	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

Specification of input or output for port 1 pins

P1DR—Port 1 Data Register **H'FFB2**

Bit	7	6	5	4	3	2	1	
	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Output data for port 1 pins

P2DR—Port 2 Data Register **H'FFB3**

Bit	7	6	5	4	3	2	1	
	P27DR	P26DR	P25DR	P24DR	P23DR	P22DR	P21DR	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Output data for port 2 pins

P3DDR—Port 3 Data Direction Register **H'FFB4**

Bit	7	6	5	4	3	2	1	
	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	
Initial value	0	0	0	0	0	0	0	
Read/Write	W	W	W	W	W	W	W	

Specification of input or output for port 3 pins

P3DR—Port 3 Data Register **H'FFB6**

Bit	7	6	5	4	3	2	1
	P37DR	P36DR	P35DR	P34DR	P33DR	P32DR	P31DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output data for port 3 pins

P4DR—Port 4 Data Register **H'FFB7**

Bit	7	6	5	4	3	2	1
	P47DR	P46DR	P45DR	P44DR	P43DR	P42DR	P41DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output data for port 4 pins

P5DDR—Port 5 Data Direction Register **H'FFB8**

Bit	7	6	5	4	3	2	1
	—	—	—	—	—	P52DDR	P51DDR
Initial value	1	1	1	1	1	0	0
Read/Write	—	—	—	—	—	W	W

Specification o
output for port

P5DR—Port 5 Data Register **H'FFBA**

Bit	7	6	5	4	3	2	1
	—	—	—	—	—	P52DR	P51DR
Initial value	1	1	1	1	1	0	0
Read/Write	—	—	—	—	—	R/W	R/W

Output data for por

P6DR—Port 6 Data Register **H'FFBB**

Bit	7	6	5	4	3	2	1
	P67DR	P66DR	P65DR	P64DR	P63DR	P62DR	P61DR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output data for port 6 pins

PBODR—Port B Output Data Register **H'FFBC**

Bit	7	6	5	4	3	2	1
	PB7ODR	PB6ODR	PB5ODR	PB4ODR	PB3ODR	PB2ODR	PB1ODR
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output data for port B pins

PBPIN—Port B Input Data Register **H'FFBD (R)**

Bit	7	6	5	4	3	2	1
	PB7PIN	PB6PIN	PB5PIN	PB4PIN	PB3PIN	PB2PIN	PB1PIN
Initial value	—*	—*	—*	—*	—*	—*	—*
Read/Write	R	R	R	R	R	R	R

Port B pin states

Note: * Determined by state of pins PB7 to PB0.

PBDDR—Port B Data Direction Register **H'FFBE (W)**

Bit	7	6	5	4	3	2	1
	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

Specification of input or output for port B pins

P7PIN—Port 7 Input Data Register **H'FFBE (R)**

Bit	7	6	5	4	3	2	1
	P77PIN	P76PIN	P75PIN	P74PIN	P73PIN	P72PIN	P71PIN
Initial value	—*	—*	—*	—*	—*	—*	—*
Read/Write	R	R	R	R	R	R	R

Port 7 pin states

Note: * Determined by state of pins P77 to P70.

P9DDR—Port 9 Data Direction Register **H'FFC0**

Bit	7	6	5	4	3	2	1
	P97DDR	P96DDR	P95DDR	P94DDR	P93DDR	P92DDR	P91DDR
Mode 1							
Initial value	0	1	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W
Modes 2 and 3							
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

Specification of input or output for port 9 pins

P9DR—Port 9 Data Register **H'FFC1**

Bit	7	6	5	4	3	2	1
	P97DR	P96DR	P95DR	P94DR	P93DR	P92DR	P91DR
Initial value	0	—*	0	0	0	0	0
Read/Write	R/W	R	R/W	R/W	R/W	R/W	R/W

Output data for port 9 pins

Note: * Determined by state of pin P96.

IRQ7 to IRQ0 enable

0	IRQn interrupt disabled
1	IRQn interrupt enabled

(n = 7 to 0)

Internal Clock Select*1

Reserved bit

Flash memory control register enable

0	Flash memory control register no
1	Flash memory control register sel

I2C master enable

0	CPU access to SCI0, SCI1, and SCI2 contr registers is enabled
1	CPU access to I2C bus interface data, PWM control registers is enabled

I2C transfer select 1 and 0*2

I2C extra buffer select

0	PA7 to PA4 are normal I/O pins
1	PA7 to PA4 are I/O pins with bus driving capability

Notes: 1. Used for 8-bit timer input clock selection. For details, see section 12.2.4, Tim Control Register (TCR).
2. Used for I2C bus interface transfer clock selection. For details, see section 1 I2C Bus Mode Register (ICMR).

0	On-chip RAM
1	On-chip RAM

Host interface enable

0	Addresses H'(FF)FFF0 to H and H'(FF)FFFC to H'(FF)F used for access to 8-bit tim X and Y) data registers and registers, and timer connec control registers
1	Addresses H'(FF)FFF0 to H and H'(FF)FFFC to H'(FF)F used for access to host inte registers and control regist keyboard controller and MO pull-up control registers

NMI edge select

0	Falling edge
1	Rising edge

External reset

0	Reset generated by watchdog timer overflow
1	Reset generated by an external reset

Interrupt control mode select

INTM1	INTM0	Description
0	0	Interrupt control mode 0
	1	Interrupt control mode 1

IOS enable

0	The $\overline{AS}$/$\overline{IOS}$ pin functions as the address strobe pin (Low output when accessing an external area)
1	The $\overline{AS}$/$\overline{IOS}$ pin functions as the I/O strobe pin (Low output when accessing a specified address from H'(FF)F000 to H'(FF)FE4F)*

Note: * In the H8S/2148 F-ZTAT A-mask version and H8S/2147 F-ZTAT A-mask version, the address range is from H'(FF)F000 to H'(FF)F7FF.

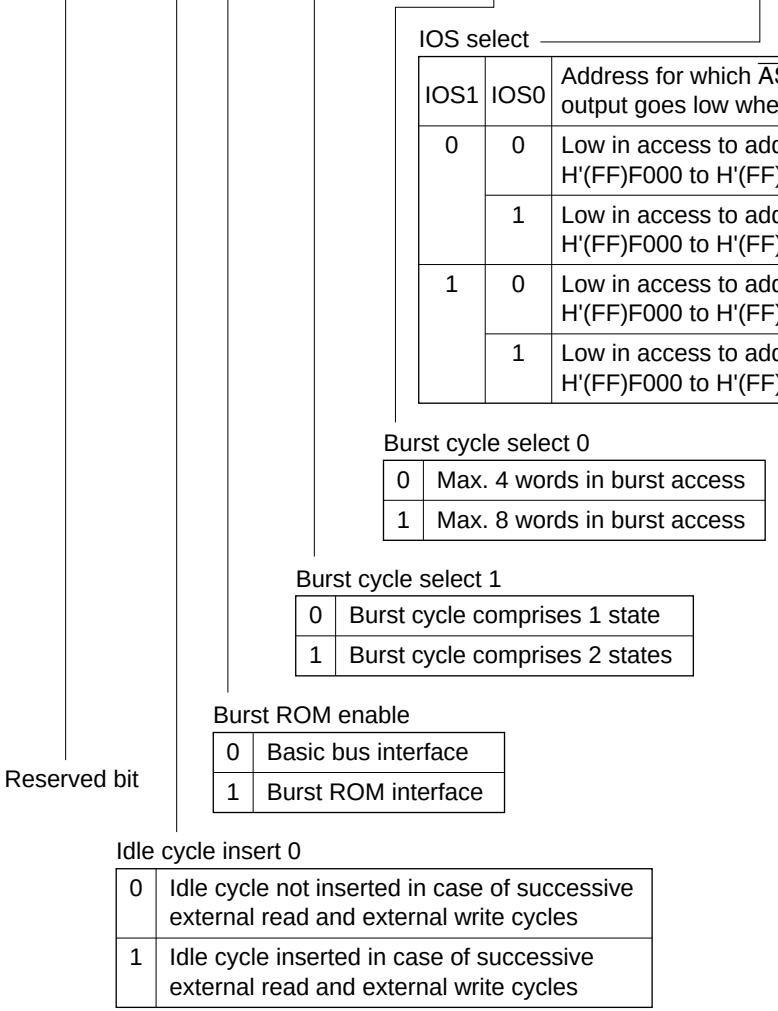
CS2 enable

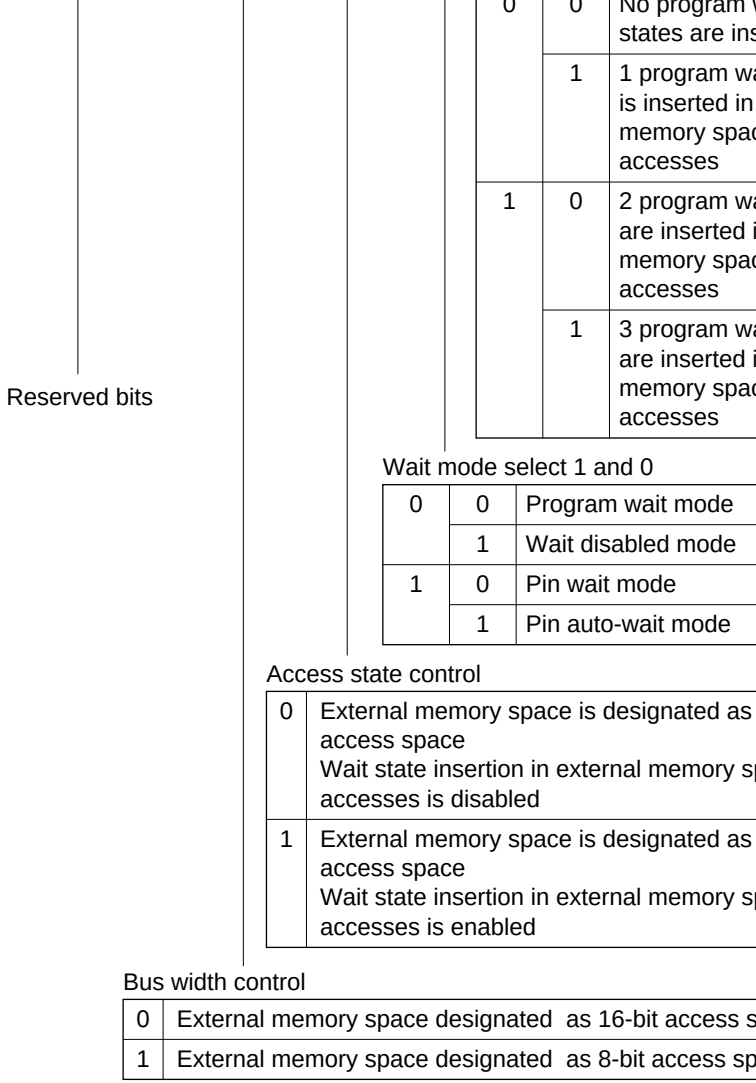
SYSCR Bit 7	HICR Bit 0	Description
CS2E	FGA20E	
0	0	$\overline{CS2}$ pin function halted ($\overline{CS2}$ fixed high internally)
	1	
1	0	$\overline{CS2}$ pin function selected for P81/$\overline{CS2}$ pin
	1	$\overline{CS2}$ pin function selected for P90/$\overline{ECS2}$ pin

Expanded mode enable

0	Single-chip mode selected
1	Expanded mode selected

Note: * Determined by the MD1 and MD0 pins.


IOS select
IOS1
IOS0
Address for which AS output goes low whe
0
0
Low in access to add H'(FF)F000 to H'(FF
1
Low in access to add H'(FF)F000 to H'(FF
1
0
Low in access to add H'(FF)F000 to H'(FF
1
Low in access to add H'(FF)F000 to H'(FF
Burst cycle select 0
0
Max. 4 words in burst access
1
Max. 8 words in burst access
Burst cycle select 1
0
Burst cycle comprises 1 state
1
Burst cycle comprises 2 states
Burst ROM enable
0
Basic bus interface
1
Burst ROM interface
Reserved bit
Idle cycle insert 0
0
Idle cycle not inserted in case of successive external read and external write cycles
1
Idle cycle inserted in case of successive external read and external write cycles



0	0	No program w states are ins
	1	1 program wa is inserted in memory spac accesses
1	0	2 program wa are inserted i memory spac accesses
	1	3 program wa are inserted i memory spac accesses

0	0	Program wait mode
	1	Wait disabled mode
1	0	Pin wait mode
	1	Pin auto-wait mode

0	External memory space is designated as access space Wait state insertion in external memory sp accesses is disabled
1	External memory space is designated as access space Wait state insertion in external memory sp accesses is enabled

0	External memory space designated as 16-bit access s
1	External memory space designated as 8-bit access sp

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Clock select 2 to 0

Channel	Bit 2	Bit 1	Bit 0	Description
	CKS2	CKS1	CKS0	
0	0	0	0	Clock input disabled
			1*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
		1	0*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
			1*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
	1	0	0	Counting at TCNT1 overflow sig
1	0	0	0	Clock input disabled
			1*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
		1	0*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
			1*1	Internal clock: counting at fallin
				Internal clock: counting at fallin
	1	0	0	Count at TCNT0 compare matc
X	0	0	0	Clock input disabled
			1	Internal clock: counting on ϕ
		1	0	Internal clock: counting at fallin
			1	Internal clock: counting at fallin
	1	0	0	Clock input disabled
Y	0	0	0	Clock input disabled
			1	Internal clock: counting at fallin
		1	0	Internal clock: counting at fallin
			1	Internal clock: counting at fallin
	1	0	0	Clock input disabled
All	1	0	1	External clock: counting at risin
		1	0	External clock: counting at fallir
			1	External clock: counting at both edges

Counter clear 1 and 0

0	0	Clear is disabled
	1	Cleared on compare match A
1	0	Cleared on compare match B
	1	Cleared on rising edge of external reset input

Timer overflow interrupt enable

0	OVF interrupt request (OVI) is disabled
1	OVF interrupt request (OVI) is enabled

Compare match interrupt enable A

0	CMFA interrupt request (CMIA) is disabled
1	CMFA interrupt request (CMIA) is enabled

Compare Match Interrupt Enable B

0	CMFB interrupt request (CMIB) is disabled
1	CMFB interrupt request (CMIB) is enabled

Notes: 1. Selected by ICKS1 and ICKS0 in STCR. For details, s
Timer Control Register (TCR).
2. If the clock input of channel 0 is the TCNT1 overflow s
channel 1 is the TCNT0 compare match signal, no inc
generated. Do not use this setting.

Output select 1 and 0

0	0	No change at compare m
	1	0 output at compare ma
1	0	1 output at compare ma
	1	Output inverted at comp match A (toggle output)

Output select 3 and 2

0	0	No change at compare match B
	1	0 output at compare match B
1	0	1 output at compare match B
	1	Output inverted at compare match B (toggle output)

A/D trigger enable

0	A/D converter start requests by compare match A are disabled
1	A/D converter start requests by compare match A are enabled

Timer overflow flag

0	[Clearing condition] Read OVF when OVF = 1, then write 0 in OVF
1	[Setting condition] When TCNT overflows from H'FF to H'00

Compare match flag A

0	[Clearing conditions] • Read CMFA when CMFA = 1, then write 0 in CMFA • When the DTC is activated by a CMIA interrupt
1	[Setting condition] When TCNT = TCORA

Compare match flag B

0	[Clearing conditions] • Read CMFB when CMFB = 1, then write 0 in CMFB • When the DTC is activated by a CMIB interrupt
1	[Setting condition] When TCNT = TCORB

Note: * Only 0 can be written in bits 7 to 5, to clear the flags.

Output select 1 and 0

0	0	No change at compare
	1	0 output at compare m
1	0	1 output at compare m
	1	Output inverted at com match A (toggle output

Output select 3 and 2

0	0	No change at compare match
	1	0 output at compare match B
1	0	1 output at compare match B
	1	Output inverted at compare match B (toggle output)

Timer overflow flag

0	[Clearing condition] Read OVF when OVF = 1, then write 0 in OVF
1	[Setting condition] When TCNT overflows from H'FF to H'00

Compare match flag A

0	[Clearing conditions] • Read CMFA when CMFA = 1, then write 0 in CMFA • When the DTC is activated by a CMIA interrupt
1	[Setting condition] When TCNT = TCORA

Compare match flag B

0	[Clearing conditions] • Read CMFB when CMFB = 1, then write 0 in CMFB • When the DTC is activated by a CMIB interrupt
1	[Setting condition] When TCNT = TCORB

Note: * Only 0 can be written in bits 7 to 5, to clear the flags.

TCORBX—Time Constant Register BX **H'FFF7**

TCORA0
TCORB0

TCORA1
TCORB1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Compare match flag (CMF) is set when TCOR and TCNT values ma

TCORAX, TCORAY
TCORBX, TCORBY

Bit	7	6	5	4	3	2	1	0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/

Compare match flag (CMF) is set when TCOR and TCNT values match

TCORC

Bit	7	6	5	4	3	2	1	0
Initial value	1	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/

Compare match C signal is generated when sum of TCORC and TICR contents match TCNT value

Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Up-counter

TCNTX, TCNTY

Bit	7	6	5	4	3	2	1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Up-counter

PWOERB	OE15	OE14	OE13	OE12	OE11	OE10	OE9
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Switching between PWM output and port output

DDR	OE	Description
0	0	Port input
	1	Port input
1	0	Port output or PWM 256/256 ou
	1	PWM output (0 to 255/256 outp

PWDPRA—PWM Data Polarity Register A **H'FFD5**
PWDPRB—PWM Data Polarity Register B **H'FFD4**

Bit	7	6	5	4	3	2	1
PWDPRA	OS7	OS6	OS5	OS4	OS3	OS2	OS1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	7	6	5	4	3	2	1
PWDPRB	OS15	OS14	OS13	OS12	OS11	OS10	OS9
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM output polarity control

0	PWM direct output (PWDR value corresponds to high width of c
1	PWM inverted output (PWDR value corresponds to low width of

0	0	0	0	PWDR0 sel
			1	PWDR1 sel
		1	0	PWDR2 sel
			1	PWDR3 sel
	1	0	0	PWDR4 sel
			1	PWDR5 sel
		1	0	PWDR6 sel
			1	PWDR7 sel
1	0	0	0	PWDR8 sel
			1	PWDR9 sel
		1	0	PWDR10 se
			1	PWDR11 se
	1	0	0	PWDR12 se
			1	PWDR13 se
		1	0	PWDR14 se
			1	PWDR15 se

PWM clock enable, PWM clock select

PWSL		PCSR		Description
Bit 7	Bit 6	Bit 2	Bit 1	
PWCKE	PWCKS	PWCKB	PWCKA	
0	—	—	—	Clock input disabled
1	0	—	—	ϕ (system clock) selected
	1	0	0	ϕ/2 selected
			1	ϕ/4 selected
		1	0	ϕ/8 selected
			1	ϕ/16 selected

ADDRAH—A/D Data Register AH	**H'FFE0**	**A/D**
ADDRAL—A/D Data Register AL	**H'FFE1**	**A/D**
ADDRBH—A/D Data Register BH	**H'FFE2**	**A/D**
ADDRBL—A/D Data Register BL	**H'FFE3**	**A/D**
ADDRCH—A/D Data Register CH	**H'FFE4**	**A/D**
ADDRCL—A/D Data Register CL	**H'FFE5**	**A/D**
ADDRDH—A/D Data Register DH	**H'FFE6**	**A/D**
ADDRDL—A/D Data Register DL	**H'FFE7**	**A/D**

Correspondence between analog input channels and ADDR registers

Analog Input Channel		A/D Data Register
Group 0	Group 1	
AN0	AN4	ADDRA
AN1	AN5	ADDRB
AN2	AN6 or CIN0 to CIN7	ADDRC
AN3	AN7 or CIN8 to CIN15	ADDRD

Channel select

Group selection	Channel selection		Description	
CH2	CH1	CH0	Single mode	Scan
0	0	0	AN0	AN0
		1	AN1	AN0, AN1
	1	0	AN2	AN0, AN1, AN
		1	AN3	AN0, AN1, AN
1	0	0	AN4	AN4
		1	AN5	AN4, AN5
	1	0	AN6 or CIN0 to 7	AN4, AN5, AN
		1	AN7 or CIN8 to 15	AN4, AN5, AN AN7 or CIN8

Clock select

0	Conversion time = 266 states (max.)
1	Conversion time = 134 states (max.)

Scan mode

0	Single mode
1	Scan mode

A/D start

0	A/D conversion stopped
1	• Single mode: A/D conversion is started. Cleared to 0 automatical when conversion on the specified channel ends • Scan mode: A/D conversion is started. Conversion continues sequentially on the selected channels until ADST is cleared to 0 by software, a reset, or a transition to standby mode or module stop mode

A/D interrupt enable

0	A/D conversion end interrupt (ADI) request disabled
1	A/D conversion end interrupt (ADI) request enabled

A/D end flag

0	[Clearing conditions] • When 0 is written in the to ADF flag after reading ADF = 1 • When the DTC is activated by an ADI interrupt, and ADDR is read
1	[Setting conditions] • Single mode: When A/D conversion ends • Scan mode: When A/D conversion ends on all specified channels

Note: * Only 0 can be written, to clear the flag.

Timer trigger select

0	0	Start of A/D conversion by external trigger is disabled
	1	Start of A/D conversion by external trigger is disabled
1	0	Start of A/D conversion by external trigger (8-bit timer) is ena
	1	Start of A/D conversion by external trigger pin is enabled

0	0	0	0
			1
		1	0
			1
	1	0	0
			1
		1	0
			1
1	0	0	0
			1
		1	0
			1
	1	0	0
			1
		1	0
			1

Reset or NMI

0	NMI interrupt requested
1	Internal reset requested

Prescaler select*2

0	TCNT counts on a ø-based prescaler (PSM) scaled clock
1	TCNT counts on a øSUB-based prescaler (PSS) scaled c

Timer enable

0	TCNT is initialized to H'00 and halted
1	TCNT counts

Timer mode select

0	Interval timer mode: Interval timer interrupt request (WOVI) sent to CPU when TCNT overflows
1	Watchdog timer mode: Reset or NMI interrupt request sent to CPU when TCNT overflows

Overflow flag

0	[Clearing conditions] • Write 0 in the TME bit • Read TCSR when OVF = 1*, then write 0 in OVF
1	[Setting condition] When TCNT overflows (changes from H'FF to H'00) (When internal reset request generation is selected in watchdog timer mode, OVF is cleared automatically by the internal reset.)

Note: * When OVF is polled and the interval timer interrupt is disabled, OVF = 1 must be read at least twice.

Notes: 1. Only 0 can be written, to clear the flag.
2. For operation control when a transition is made to power-down mode, see section 25.2.3, Timer Control/Status Register (

Fast gate A20 enabl

0	Fast gate A20 f disabled
1	Fast gate A20 f enabled

Input data register full interrupt

0	Input data register (IDR1) receive complete interrupt disabled
1	Input data register (IDR1) receive complete interrupt enabled

Input data register full interrupt enable 2

0	Input data register (IDR2) receive comp interrupt is disabled
1	Input data register (IDR2) receive comp interrupt is enabled

0	0	No change at compare
	1	0 output at compare m
1	0	1 output at compare m
	1	Output inverted at com match A (toggle output

Output select 3 and 2

0	0	No change at compare matc
	1	0 output at compare match B
1	0	1 output at compare match B
	1	Output inverted at compare match B (toggle output)

Input capture flag

0	[Clearing condition] When 0 is written in ICF after reading ICF = 1
1	[Setting condition] When a rising edge followed by a falling edge detected in the external reset signal after the ICST bit in TCONRI has been set to 1

Timer overflow flag

0	[Clearing condition] When 0 is written in OVF after reading OVF = 1
1	[Setting condition] When TCNT overflows from H'FF to H'00

Compare match flag A

0	[Clearing conditions] • When 0 is written in CMFA after reading CMFA = 1 • When the DTC is activated by a CMIA interrupt
1	[Setting condition] When TCNT = TCORA

Compare match flag B

0	[Clearing conditions] • When 0 is written in CMFB after reading CMFB = 1 • When the DTC is activated by a CMIB interrupt
1	[Setting condition] When TCNT = TCORB

Note: * Only 0 can be written in bits 7 to 4, to clear the flags.

Output select 1 and 0

0	0	No change at compare
	1	0 output at compare m
1	0	1 output at compare m
	1	Output inverted at com match A (toggle output

Output select 3 and 2

0	0	No change at compare match
	1	0 output at compare match B
1	0	1 output at compare match B
	1	Output inverted at compare match B (toggle output)

Input capture interrupt enable

0	Interrupt request by ICF (ICIX) is disabled
1	Interrupt request by ICF (ICIX) is enabled

Timer overflow flag

0	[Clearing condition] When 0 is written in OVF after reading OVF = 1
1	[Setting condition] When TCNT overflows from H'FF to H'00

Compare match flag A

0	[Clearing conditions] • When 0 is written in CMFA after reading CMFA = 1 • When the DTC is activated by a CMIA interrupt
1	[Setting condition] When TCNT = TCORA

Compare match flag B

0	[Clearing conditions] • When 0 is written in CMFB after reading CMFB = 1 • When the DTC is activated by a CMIB interrupt
1	[Setting condition] When TCNT = TCORB

Note: * Only 0 can be written in bits 7 to 5, to clear the flags.

Keyboard matrix interrupt mask

0	Key-sense input interrupt requests enable
1	Key-sense input interrupt requests disable

KMIMRA

Bit	7	6	5	4	3	2	1
	KMIMR15	KMIMR14	KMIMR13	KMIMR12	KMIMR11	KMIMR10	KMIMR9
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Keyboard matrix interrupt mask

0	Key-sense input interrupt requests enable
1	Key-sense input interrupt requests disable

TICRR—Input Capture Register R **H'FFF2**
TICRF—Input Capture Register F **H'FFF3**

Bit	7	6	5	4	3	2	1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

Stores TCNT value at fall of external reset input

Note: KMPCR has the same address as TICRR/TCORAY of TMRX/TMRY. When selecting KMPCR, set the HIE bit to 1 in SYSCR.

IDR1—Input Data Register 1 **H'FFF4**
IDR2—Input Data Register 2 **H'FFFC**

Bit	7	6	5	4	3	2	1
	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1
Initial value	—	—	—	—	—	—	—
Slave R/W	R	R	R	R	R	R	R
Host R/W	W	W	W	W	W	W	W

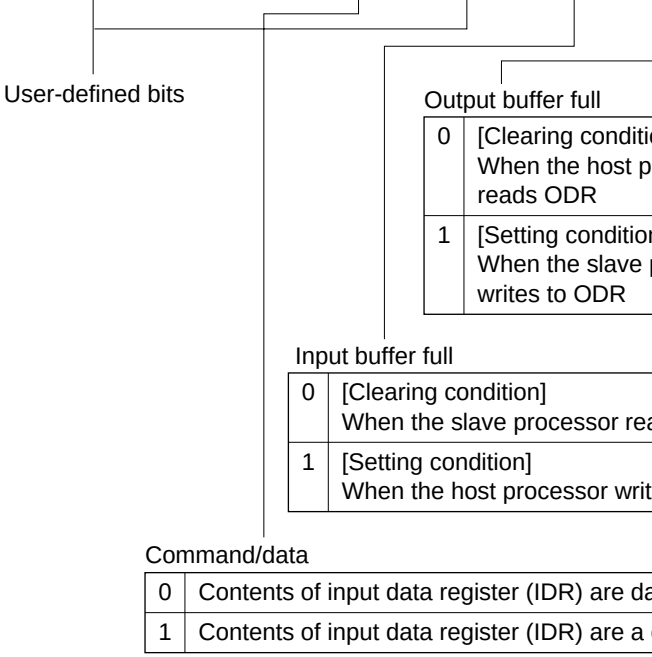
Stores host data bus contents at rise of $\overline{IOW}$ when $\overline{CS}$ is low

ODR1—Output Data Register 1 **H'FFF5**
ODR2—Output Data Register 2 **H'FFFD**

Bit	7	6	5	4	3	2	1
	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1
Initial value	—	—	—	—	—	—	—
Slave R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Host R/W	R	R	R	R	R	R	R

ODR contents are output to the host data bus when HA0 is low, $\overline{CS}$ is low, and $\overline{IOR}$ is low

0	IVG signal is selected (H8S/2148 Group) External clock/reset input is disabled (H8S/214 H8S/2147N)
1	VSYNCI/TMIY (TMCIY/TMRIY) is selected



DADR0—D/A Data Register 0 **H'FFF8** **D/A**
DADR1—D/A Data Register 1 **H'FFF9** **D/A**

Bit	7	6	5	4	3	2	1	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Stores data for D/A conversion

DAOE1	DAOE0	DAE	Conversion result
0	0	*	Channel 0 and 1 D/A conversi
	1	0	Channel 0 D/A conversion ena Channel 1 D/A conversion dis
		1	Channel 0 and 1 D/A conversi
1	0	0	Channel 0 D/A conversion disa Channel 1 D/A conversion ena
		1	Channel 0 and 1 D/A conversi
	1	*	Channel 0 and 1 D/A conversi

Legend: *: Don't care

D/A output enable 0

0	Analog output DA0 disabled
1	D/A conversion is enabled on channel 0. Analog output DA0 is enabled

D/A output enable 1

0	Analog output DA1 disabled
1	D/A conversion is enabled on channel 1. Analog output DA1 is enabled

	is used direc the VSYNCI
1	The VSYNC is inverted b as the VSYN

Input synchronization signal inversion

0	The HSYNCI and CSYNCI pin states ar directly as the HSYNCI and CSYNCI inp
1	The HSYNCI and CSYNCI pin states ar before use as the HSYNCI and CSYNC

Input synchronization signal inversion

0	The VFBACKI pin state is used directly as the VFBACKI in
1	The VFBACKI pin state is inverted before use as the VFBA

Input synchronization signal inversion

0	The HFBACKI pin state is used directly as the HFBACKI input
1	The HFBACKI pin state is inverted before use as the HFBACK

Input capture start bit

0	The TICRR and TICRF input capture functions are stopped [Clearing condition] When a rising edge followed by a falling edge is detected on TMRIX
1	The TICRR and TICRF input capture functions are operating (Waiting for detection of a rising edge followed by a falling edge on [Setting condition] When 1 is written in ICST after reading ICST = 0

Synchronization signal connection enable

SCONE	Mode	FTIA	FTIB	FTIC	FTID	TMCI1
0	Normal connection	FTIA input	FTIB input	FTIC input	FTID input	TMCI1 input
1	Synchronization signal connec-tion mode	IVI signal	TMO1 signal	VFBACKI input	IHI signal	IHI signal

Input synchronization mode select 1 and 0

SIMOD1	SIMOD0	Mode	IHI signal	IVI signal
0	0	No signal	HFBACKI input	VFBACKI input
	1	S-on-G mode	CSYNCI input	PDC input
1	0	Composite mode	HSYNCI input	PDC input
	1	Separate mode	HSYNCI input	VSYNCI input

Signal inversion

0	The CBLA used direc CBLANK o
1	The CBLA inverted be the CBLAN

Output synchronization sig

0	The CLO signal (CL1 or CL4 signal) is use the CLAMPO output
1	The CLO signal (CL1 or CL4 signal) is inve use as the CLAMPO

Output synchronization signal invers

0	The IVO signal is used directly the VSYNCO output
1	The IVO signal is inverted befo use as the VSYNCO output

Output synchronization signal inversion

0	The IHO signal is used directly as the HSYNCO o
1	The IHO signal is inverted before use as the HSY

Output enable

0	The P27/A15/PW15/CBLANK pin functions as the P27/A15
1	In mode 1 (expanded mode with on-chip ROM disabled): The P27/A15/PW15/CBLANK pin functions as the A15 p In modes 2 and 3 (expanded modes with on-chip ROM ena The P27/A15/PW15/CBLANK pin functions as the CBLA

Output enable

0	The P64/FTIC/$\overline{\text{KIN4}}$/CIN4/CLAMPO pin functions as the P64/FTIC/$\overline{\text{KIN}}$
1	The P64/FTIC/$\overline{\text{KIN4}}$/CIN4/CLAMPO pin functions as the CLAMPO pin

Output enable

0	The P61/FTOA/$\overline{\text{KIN1}}$/CIN1/VSYNCO pin functions as the P61/FTOA/$\overline{\text{KIN1}}$/CIN1 pi
1	The P61/FTOA/$\overline{\text{KIN1}}$/CIN1/VSYNCO pin functions as the VSYNCO pin

Output enable

0	The P44/TMO1/HIRQ1/HSYNCO pin functions as the P44/TMO1/HIRQ1 pin
1	The P44/TMO1/HIRQ1/HSYNCO pin functions as the HSYNCO pin

0	0	0	The CL1 signal is
		1	The CL2 signal is
	1	0	The CL3 signal is
		1	
1	0	0	The CL4 signal is
		1	
	1	0	
		1	

Vertical synchronization output mode select 1 and 0

ISGENE	VOMOD1	VOMOD0	Description
0	0	0	The IVI signal (without fall modifi or IHI synchronization) is selecte
		1	The IVI signal (without fall modifi with IHI synchronization) is selec
	1	0	The IVI signal (with fall modificati without IHI synchronization) is se
		1	The IVI signal (with fall modificati IHI synchronization) is selected
1	0	0	The IVG signal is selected
		1	
	1	0	
		1	

Horizontal synchronization output mode select 1 and 0

ISGENE	HOMOD1	HOMOD0	Description
0	0	0	The IHI signal (without 2fH modification) is
		1	The IHI signal (with 2fH modification) is sele
	1	0	The CL1 signal is selected
		1	
1	0	0	The IHG signal is selected
		1	
	1	0	
		1	

Internal synchronization signal select

TMRX/TMRY access select

0	The TMRX registers are accessed at addresses H'FFF0 to H'FFF5
1	The TMRY registers are accessed at addresses H'FFF0 to H'FFF5

1	The IVI sig

IHI signal level

0	The IHI signal is
1	The IHI signal is

Pre-equalization flag

0	[Clearing condition] When 0 is written in PRE reading PREQF = 1
1	[Setting condition] When an IHI signal 2fH r condition is detected

VFBACKI edge

0	[Clearing condition] When 0 is written in VFEDG after reading
1	[Setting condition] When a rising edge is detected on the VF

HFBACKI edge

0	[Clearing condition] When 0 is written in HFEDG after reading HFED
1	[Setting condition] When a rising edge is detected on the HFBACK

CSYNCI edge

0	[Clearing condition] When 0 is written in CEDG after reading CEDG = 1
1	[Setting condition] When a rising edge is detected on the CSYNCI pin

HSYNCI edge

0	[Clearing condition] When 0 is written in HEDG after reading HEDG = 1
1	[Setting condition] When a rising edge is detected on the HSYNCI pin

VSYNCI edge

0	[Clearing condition] When 0 is written in VEDG after reading VEDG = 1
1	[Setting condition] When a rising edge is detected on the VSYNCI pin

Notes: 1. Only 0 can be written, to clear the flags.
2. The initial value is undefined since it depends on the pin states.

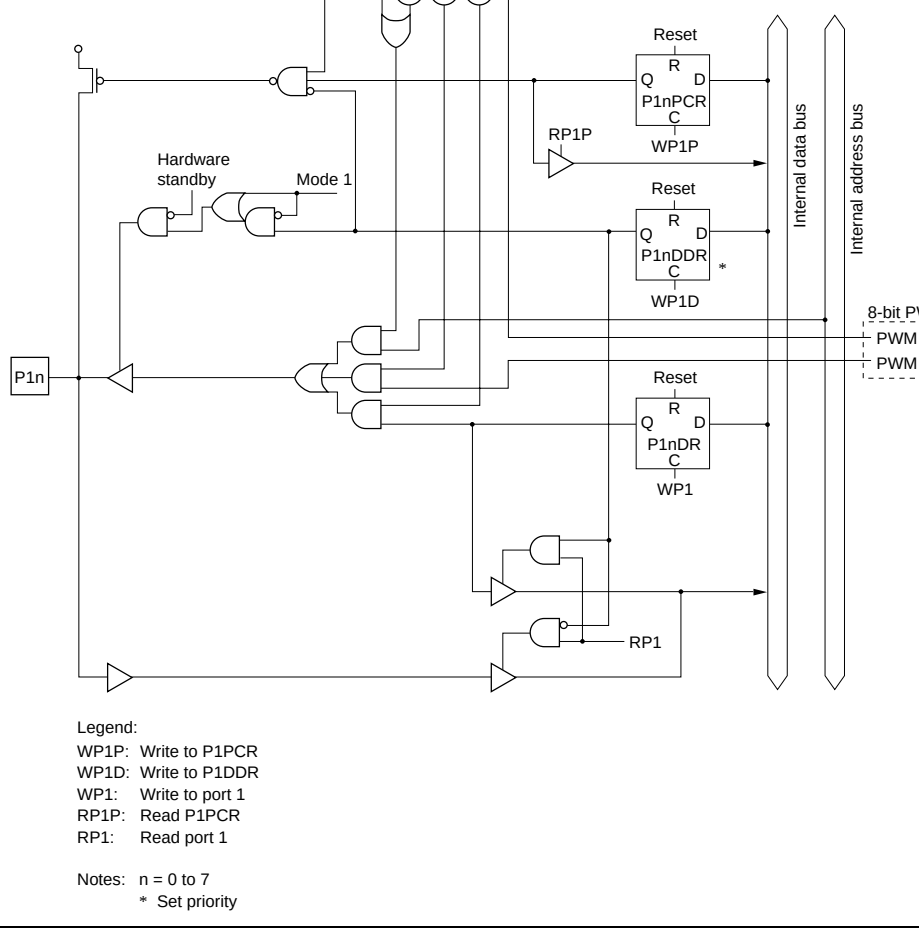
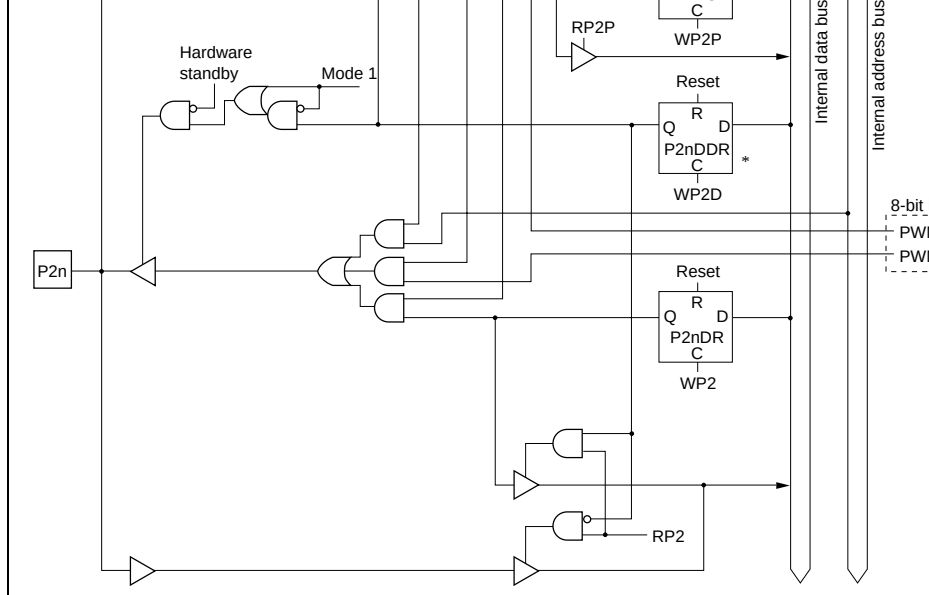


Figure C.1 Port 1 Block Diagram



Legend:
WP2P: Write to P2PCR
WP2D: Write to P2DDR
WP2: Write to port 2
RP2P: Read P2PCR
RP2: Read port 2

Notes: n = 0 to 3
* Set priority

Figure C.2 Port 2 Block Diagram (Pins P20 to P23)

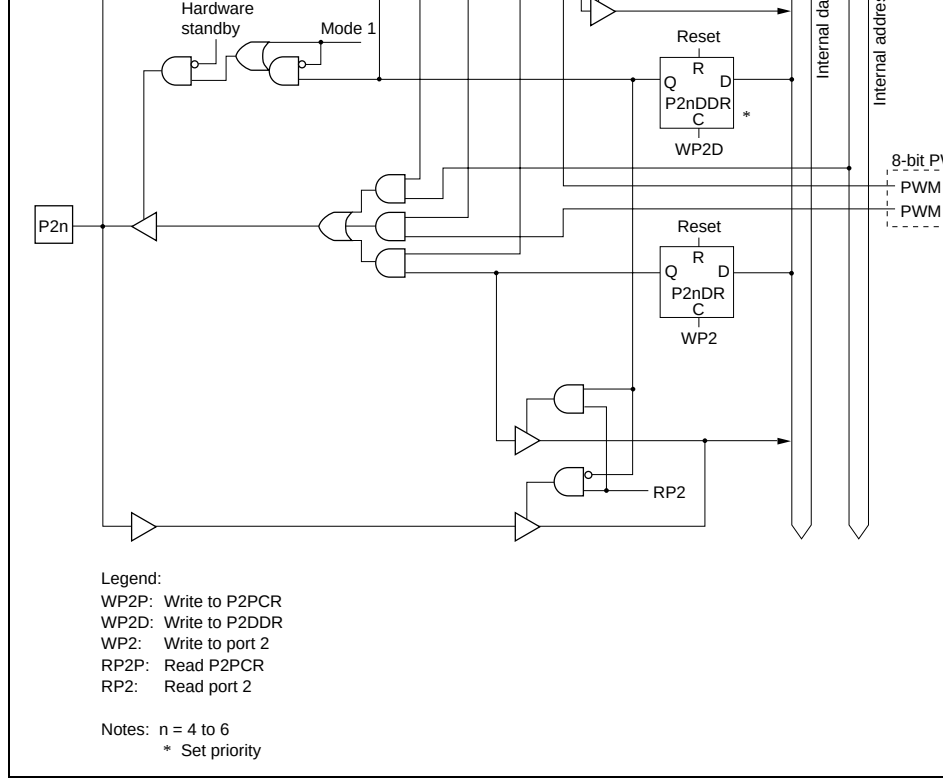
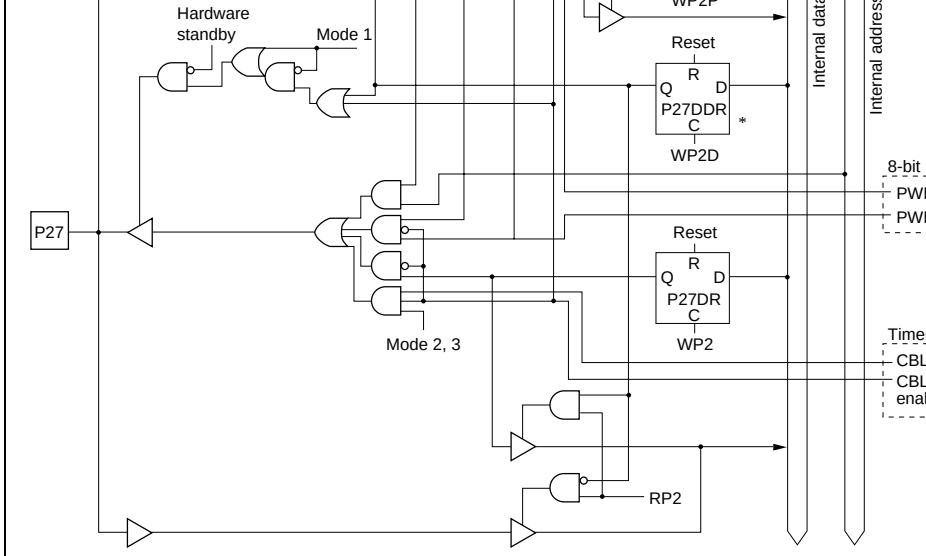


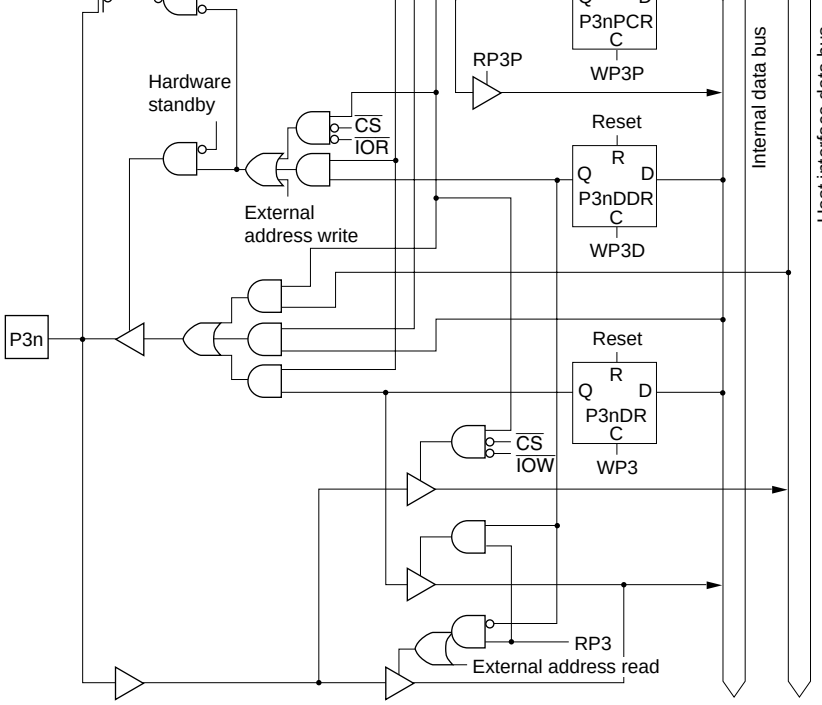
Figure C.3 Port 2 Block Diagram (Pins P24 to P26)



Legend:
WP2P: Write to P2PCR
WP2D: Write to P2DDR
WP2: Write to port 2
RP2P: Read P2PCR
RP2: Read port 2

Note: * Set priority

Figure C.4 Port 2 Block Diagram (Pin P27)



Legend:
WP3P: Write to P3PCR
WP3D: Write to P3DDR
WP3: Write to port 3
RP3P: Read P3PCR
RP3: Read port 3

Note: n = 0 to 7

Figure C.5 Port 3 Block Diagram

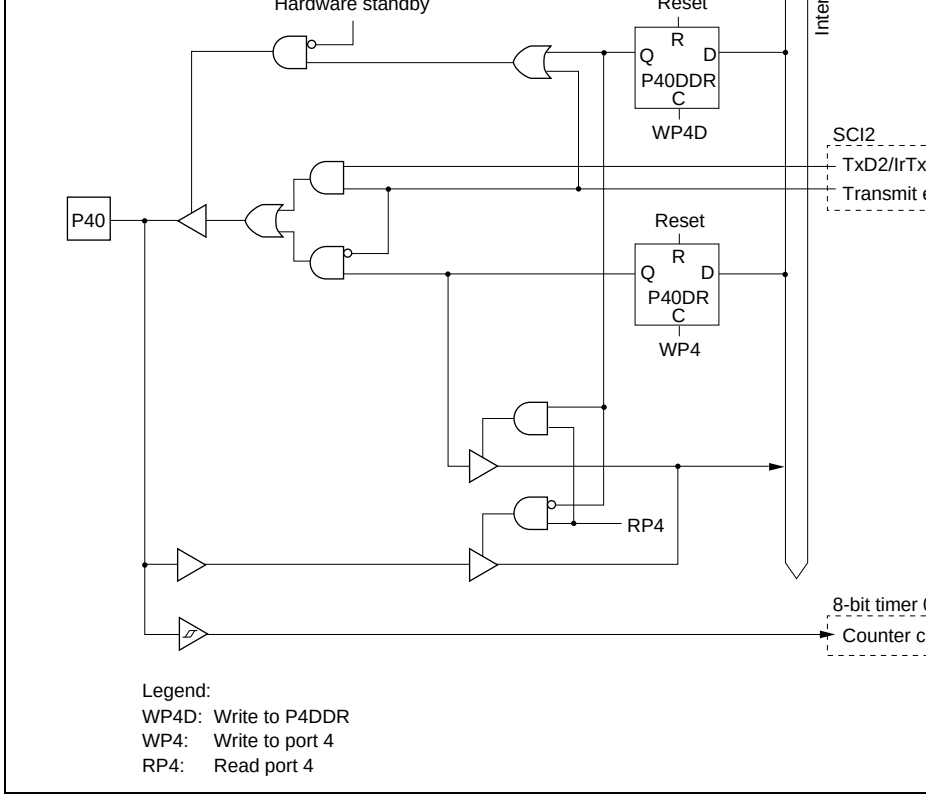


Figure C.6 Port 4 Block Diagram (Pin P40)

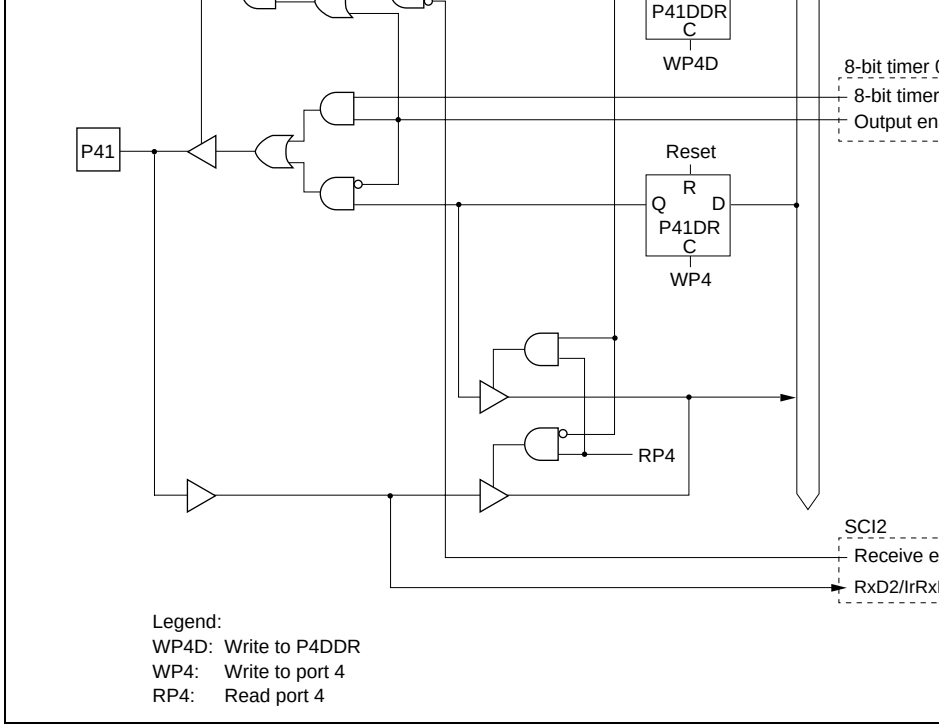


Figure C.7 Port 4 Block Diagram (Pin P41)

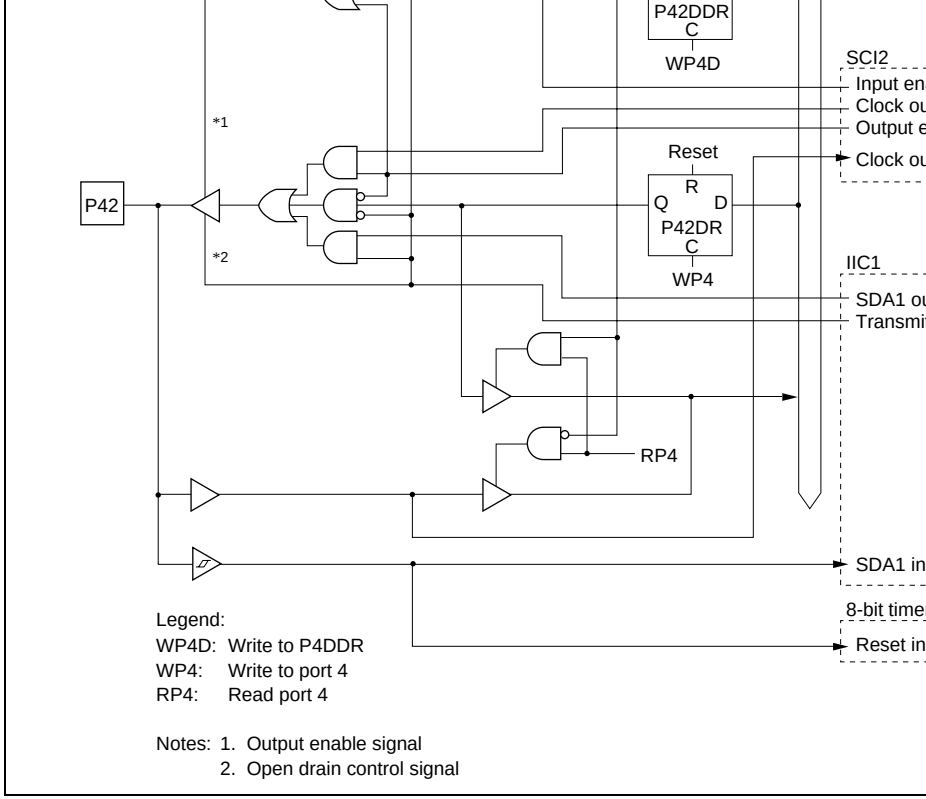


Figure C.8 Port 4 Block Diagram (Pin P42)

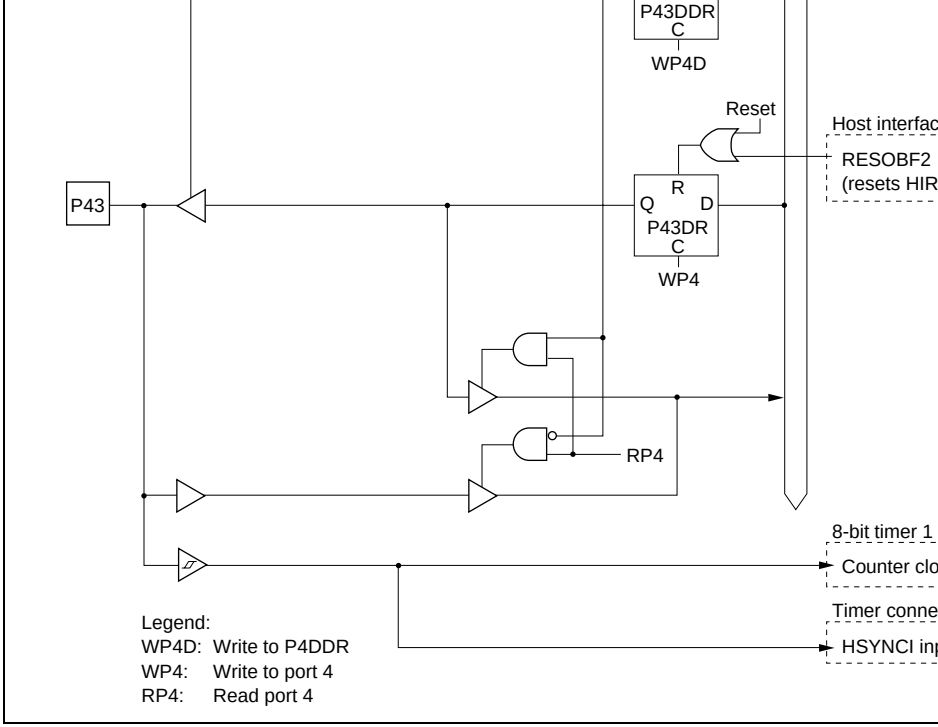


Figure C.9 Port 4 Block Diagram (Pin P43)

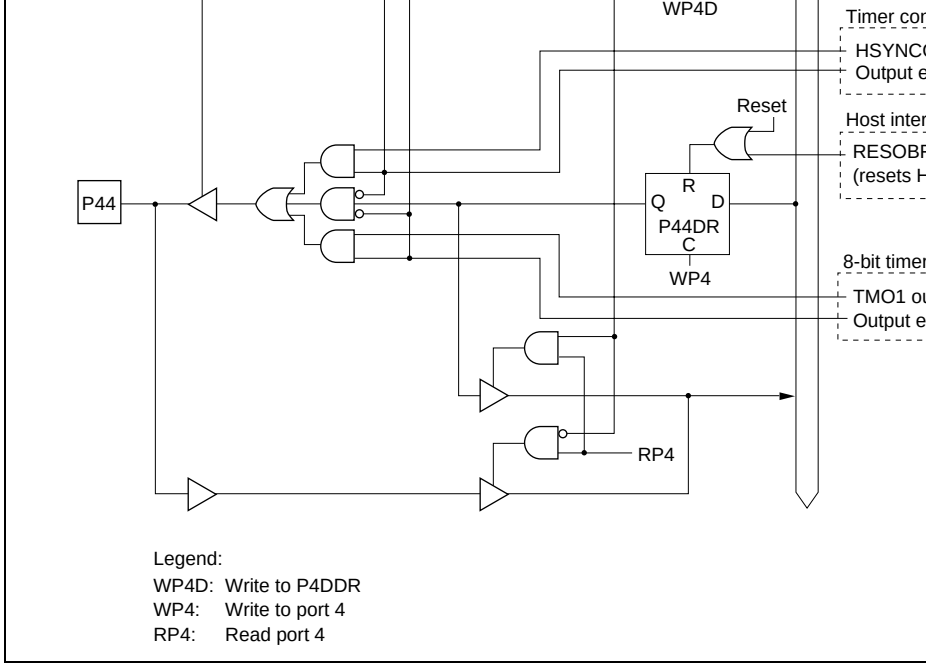


Figure C.10 Port 4 Block Diagram (Pin P44)

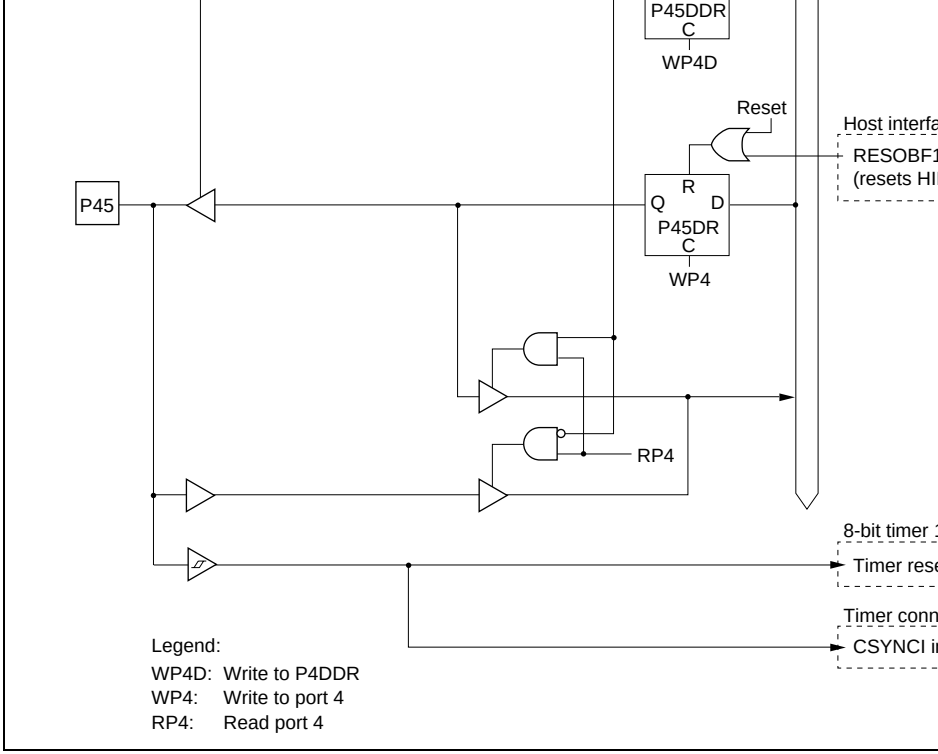
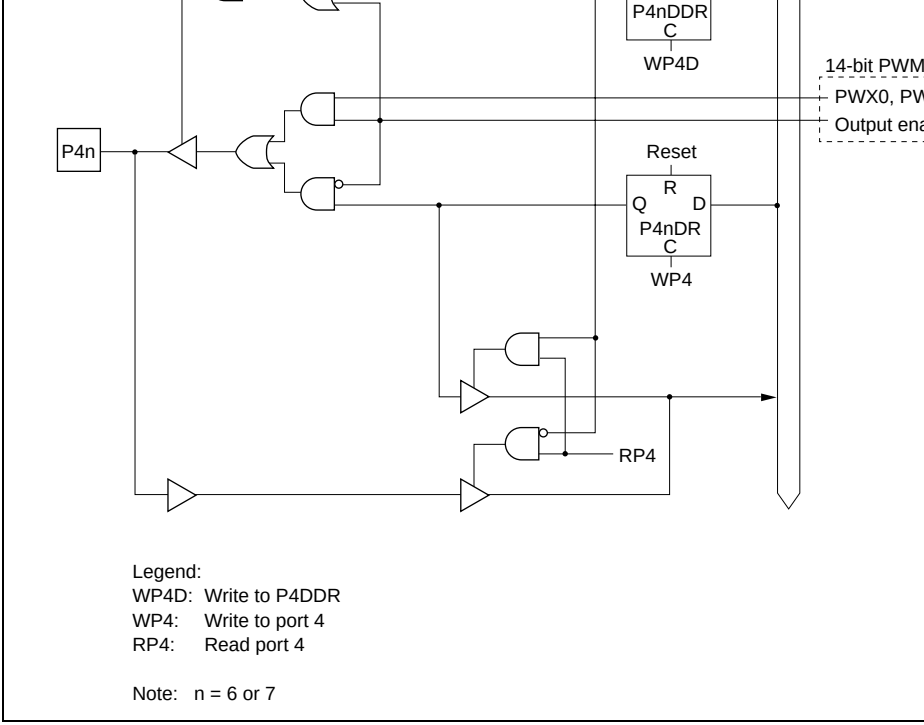


Figure C.11 Port 4 Block Diagram (Pin P45)



Legend:
WP4D: Write to P4DDR
WP4: Write to port 4
RP4: Read port 4

Note: n = 6 or 7

Figure C.12 Port 4 Block Diagram (Pins P46, P47)

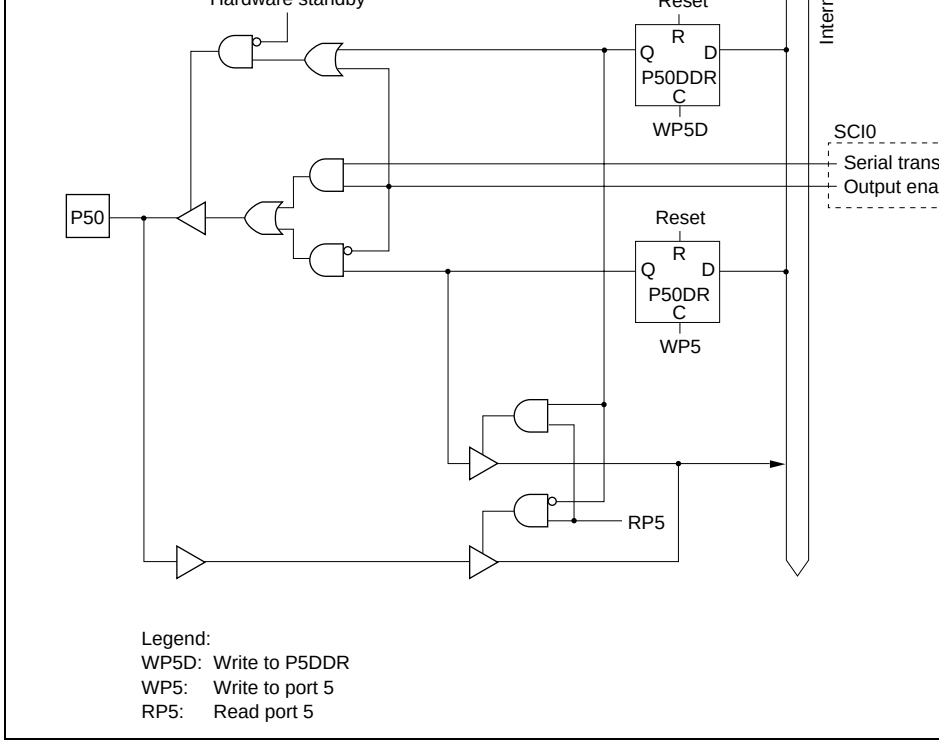


Figure C.13 Port 5 Block Diagram (Pin P50)

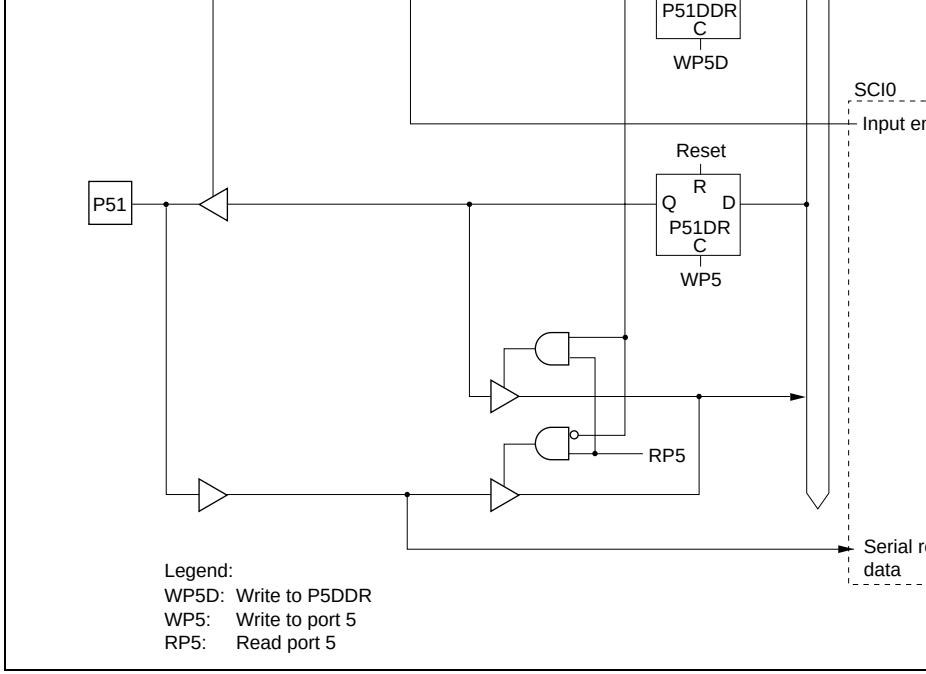
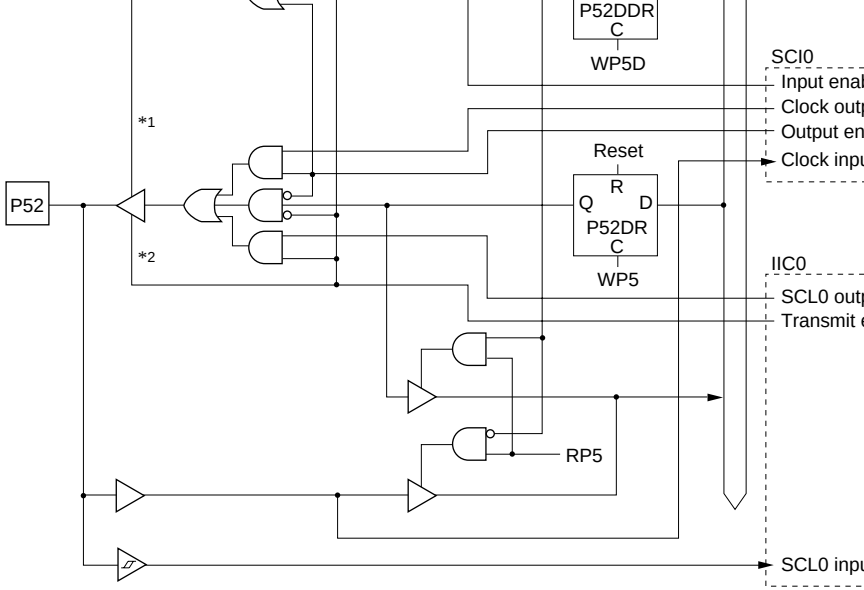


Figure C.14 Port 5 Block Diagram (Pin P51)



Legend:
WP5D: Write to P5DDR
WP5: Write to port 5
RP5: Read port 5

Notes: 1. Output enable signal
2. Open drain control signal

Figure C.15 Port 5 Block Diagram (Pin P52)

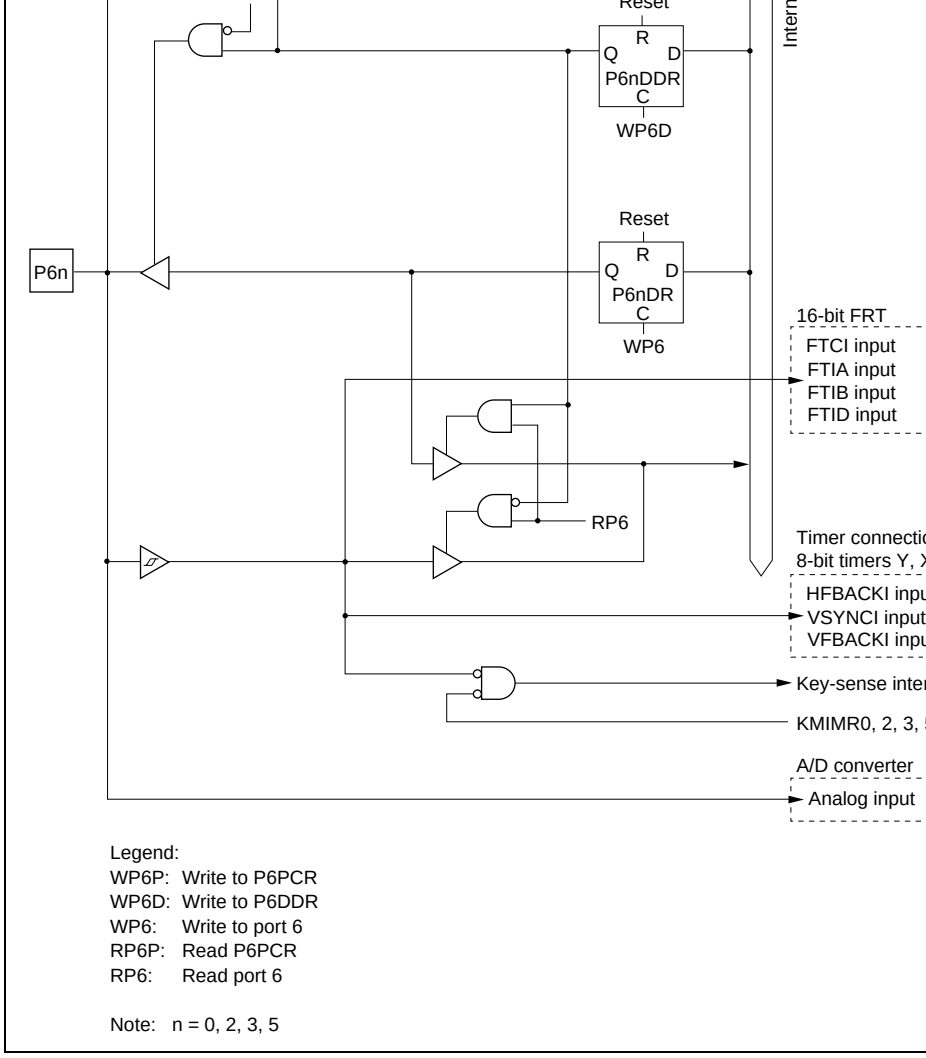
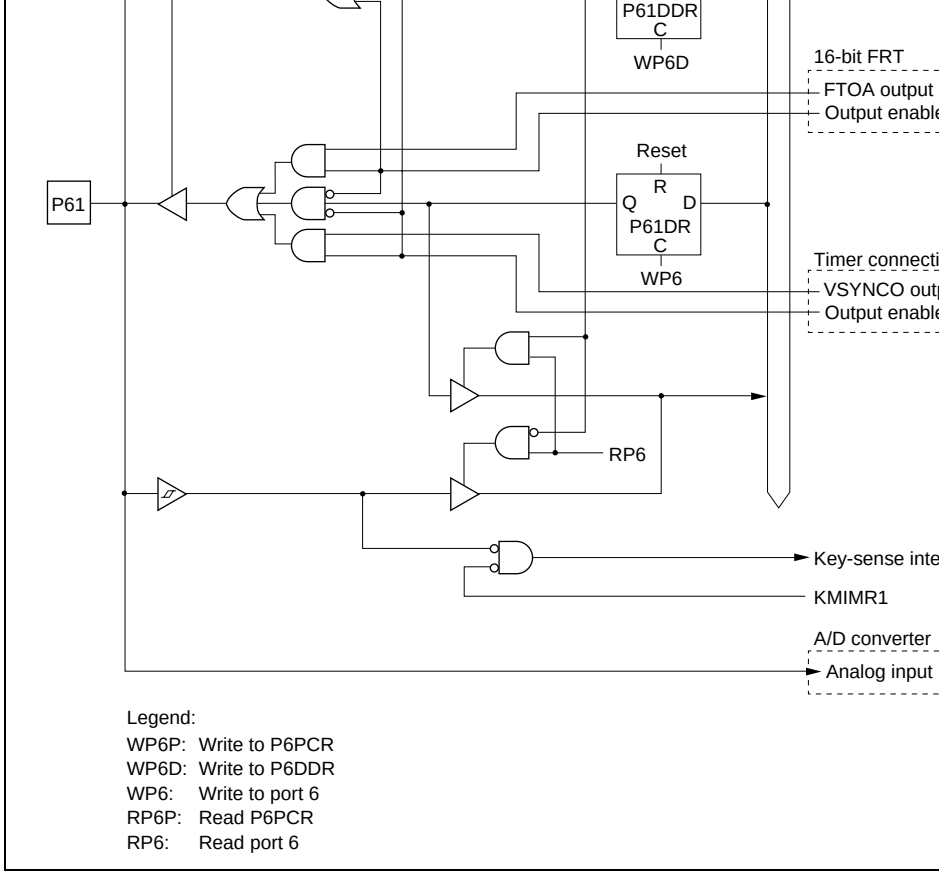


Figure C.16 Port 6 Block Diagram (Pins P60, P62, P63, P65)



Legend:
WP6P: Write to P6PCR
WP6D: Write to P6DDR
WP6: Write to port 6
RP6P: Read P6PCR
RP6: Read port 6

Figure C.17 Port 6 Block Diagram (Pin P61)

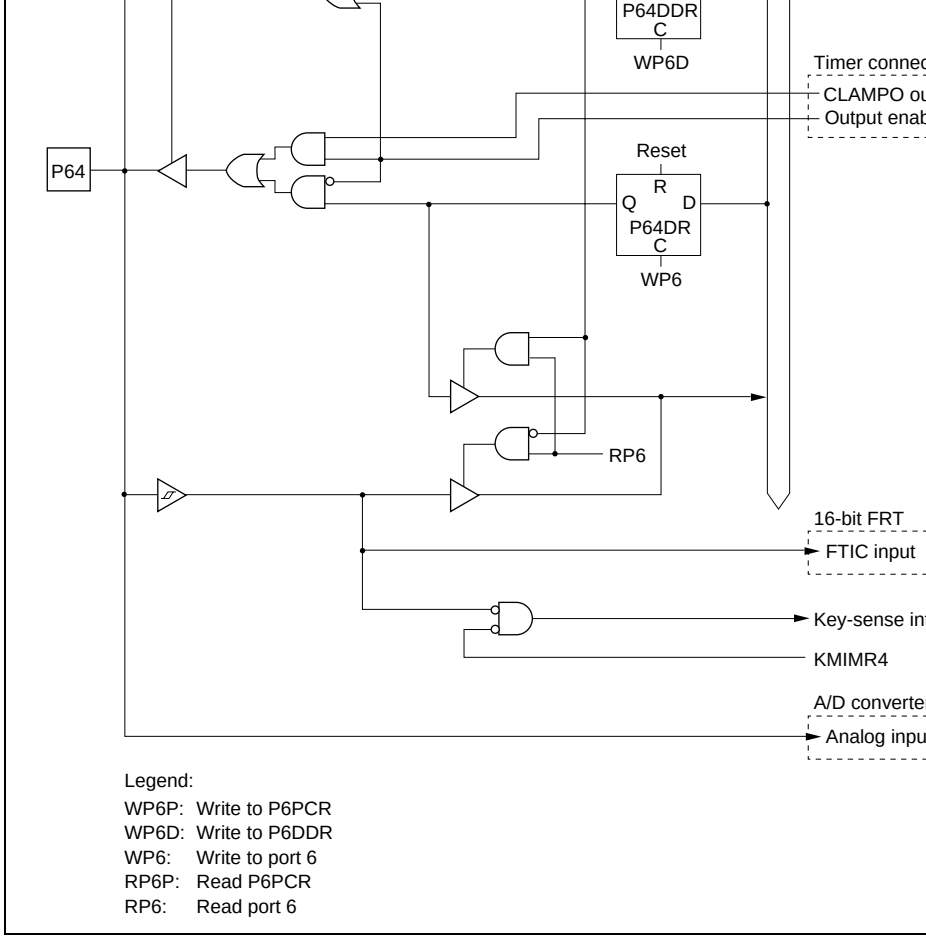


Figure C.18 Port 6 Block Diagram (Pin P64)

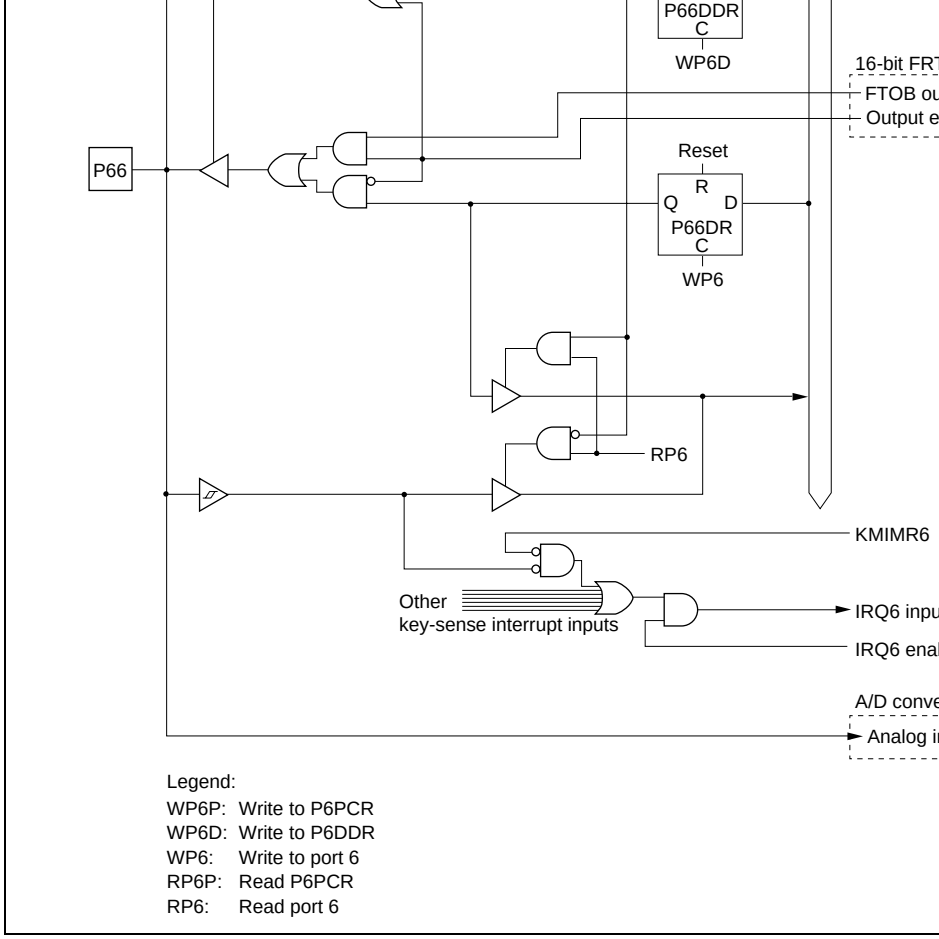


Figure C.19 Port 6 Block Diagram (Pin P66)

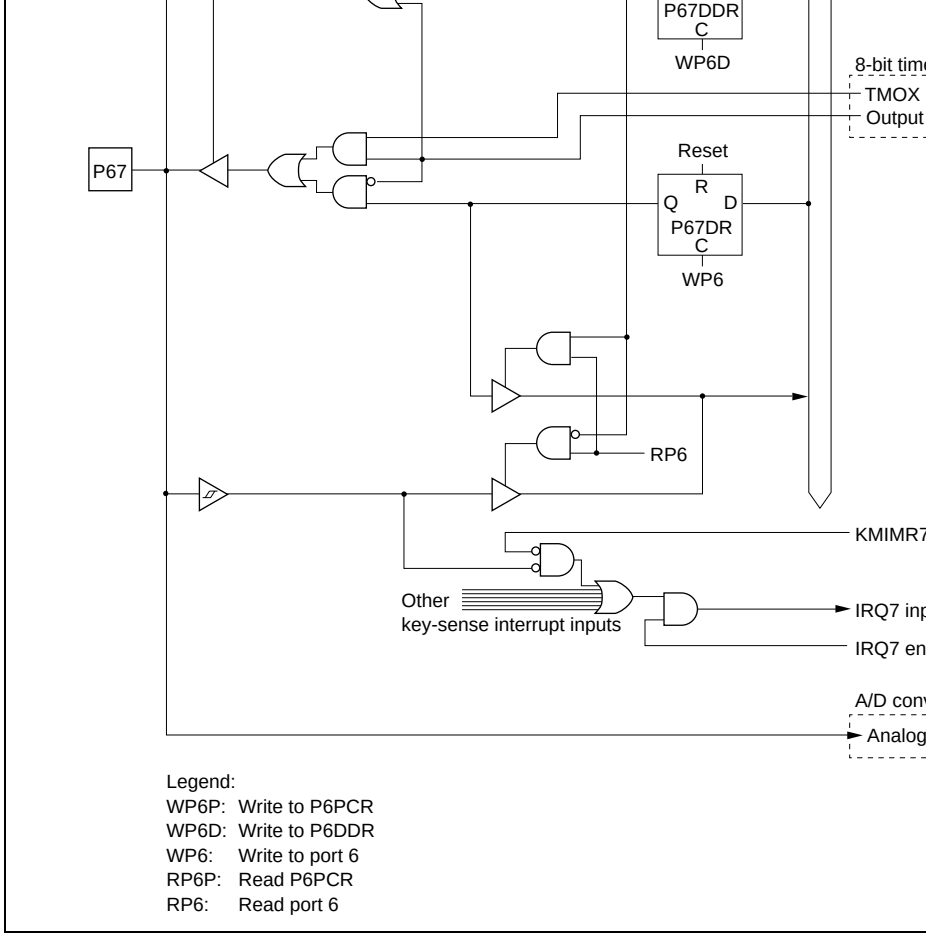
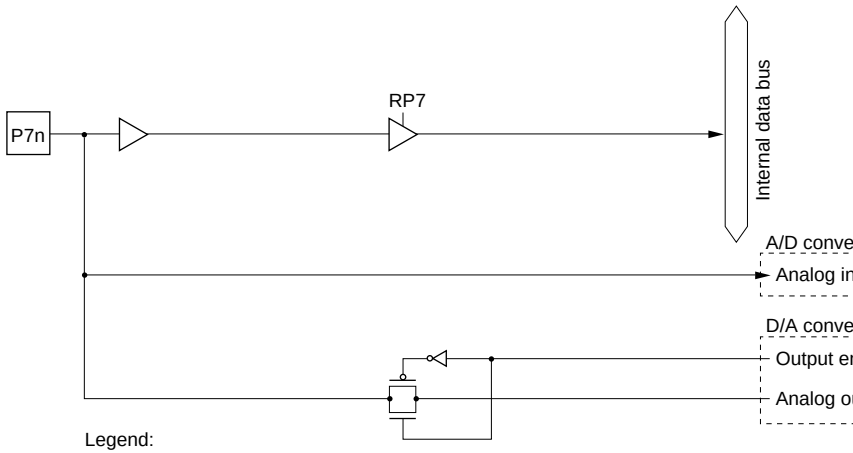


Figure C.20 Port 6 Block Diagram (Pin P67)

Legend:
RP7: Read port 7

Note: n = 0 to 5

Figure C.21 Port 7 Block Diagram (Pins P70 to P75)



Legend:
RP7: Read port 7

Note: n = 6 or 7

Figure C.22 Port 7 Block Diagram (Pins P76, P77)

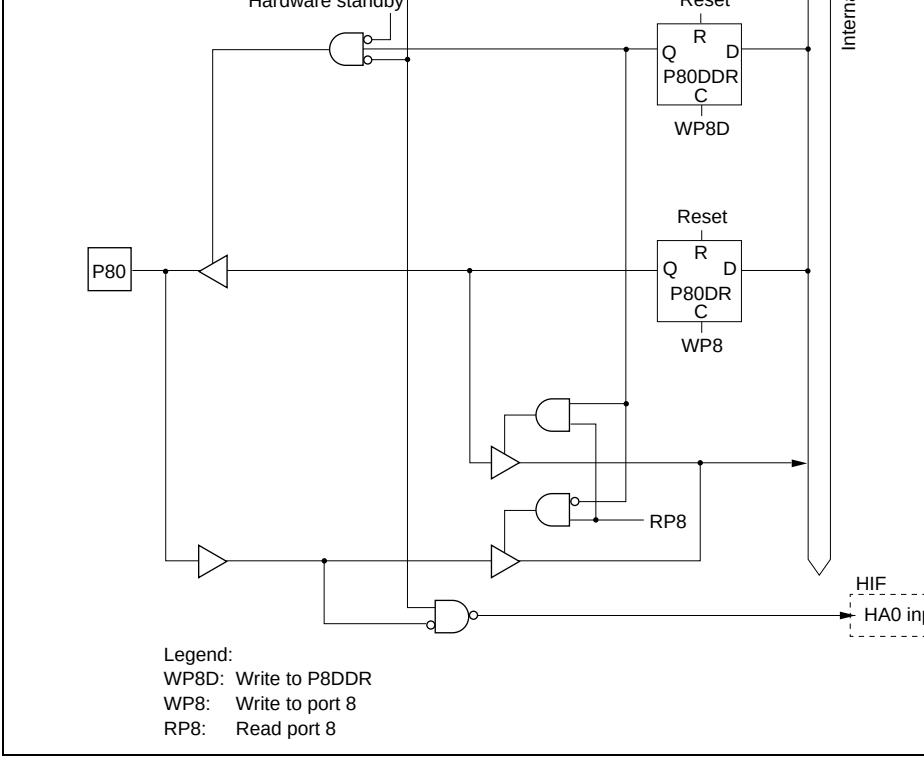


Figure C.23 Port 8 Block Diagram (Pin P80)

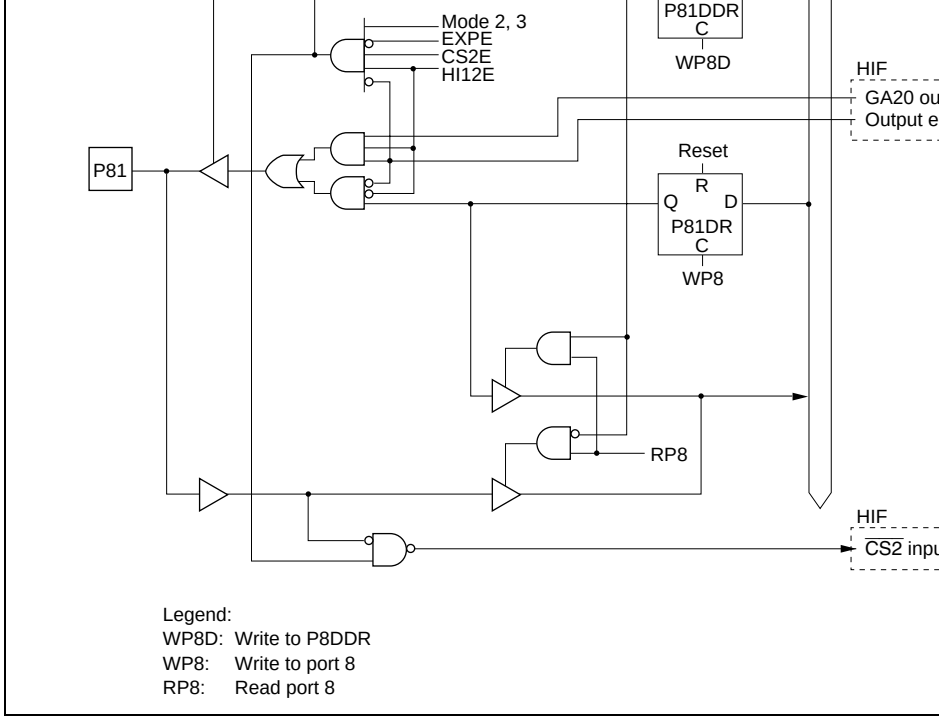


Figure C.24 Port 8 Block Diagram (Pin P81)

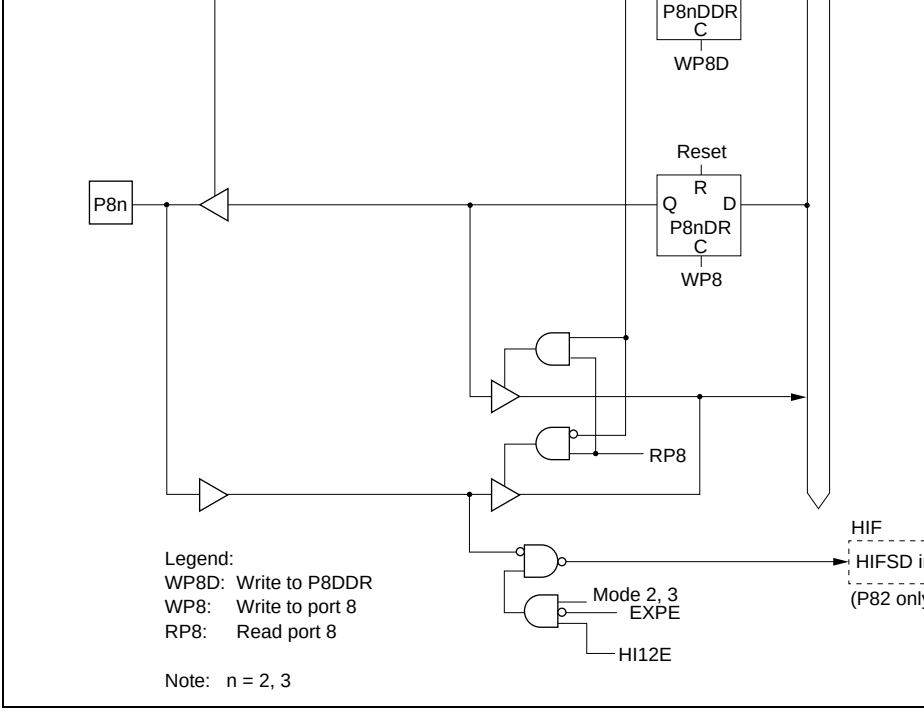


Figure C.25 Port 8 Block Diagram (Pins P82, P83)

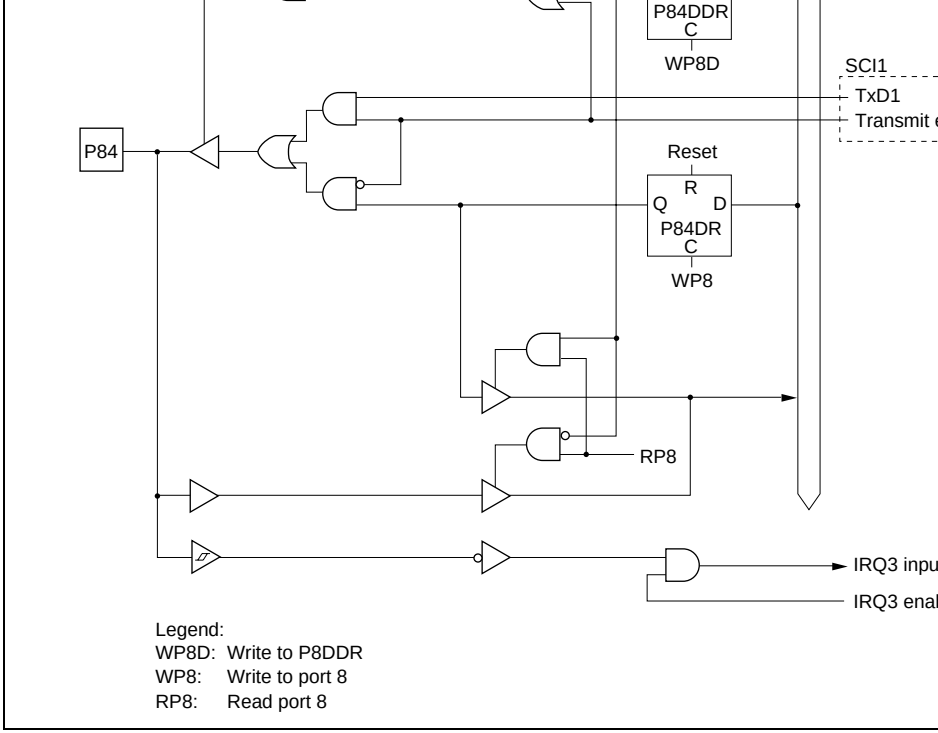


Figure C.26 Port 8 Block Diagram (Pin P84)

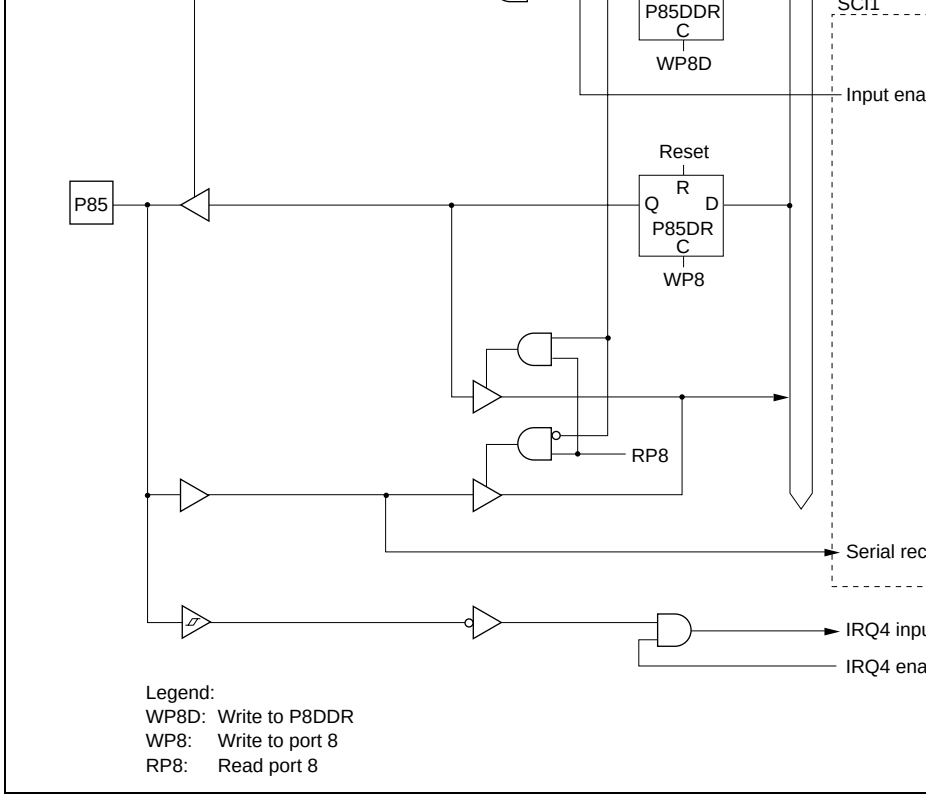


Figure C.27 Port 8 Block Diagram (Pin P85)

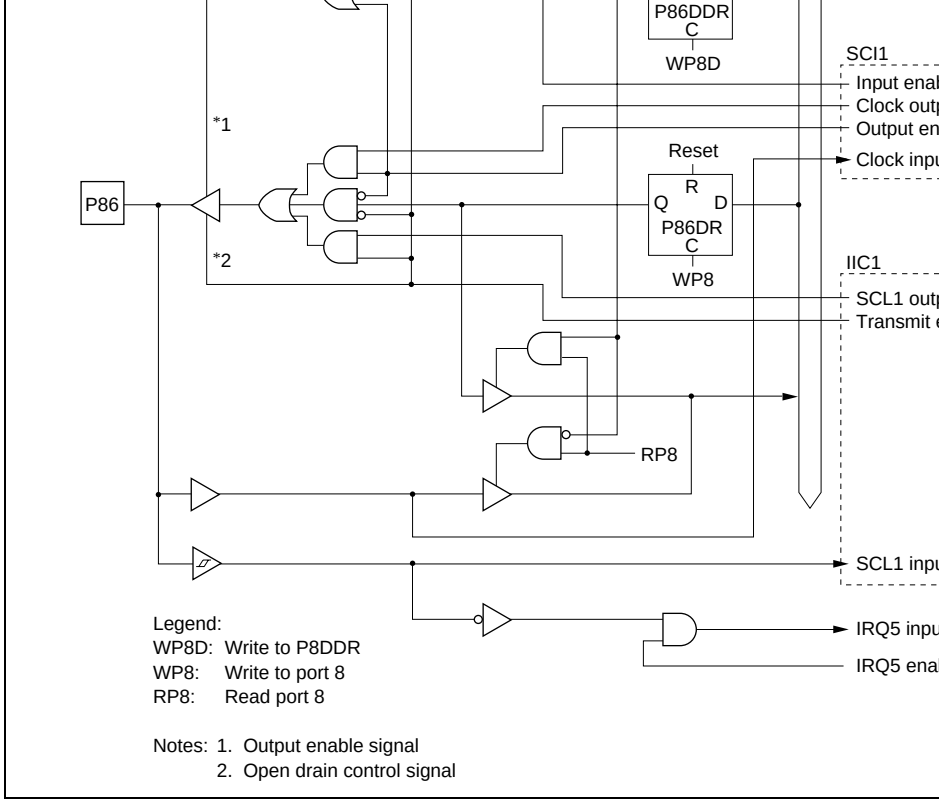


Figure C.28 Port 8 Block Diagram (Pin P86)

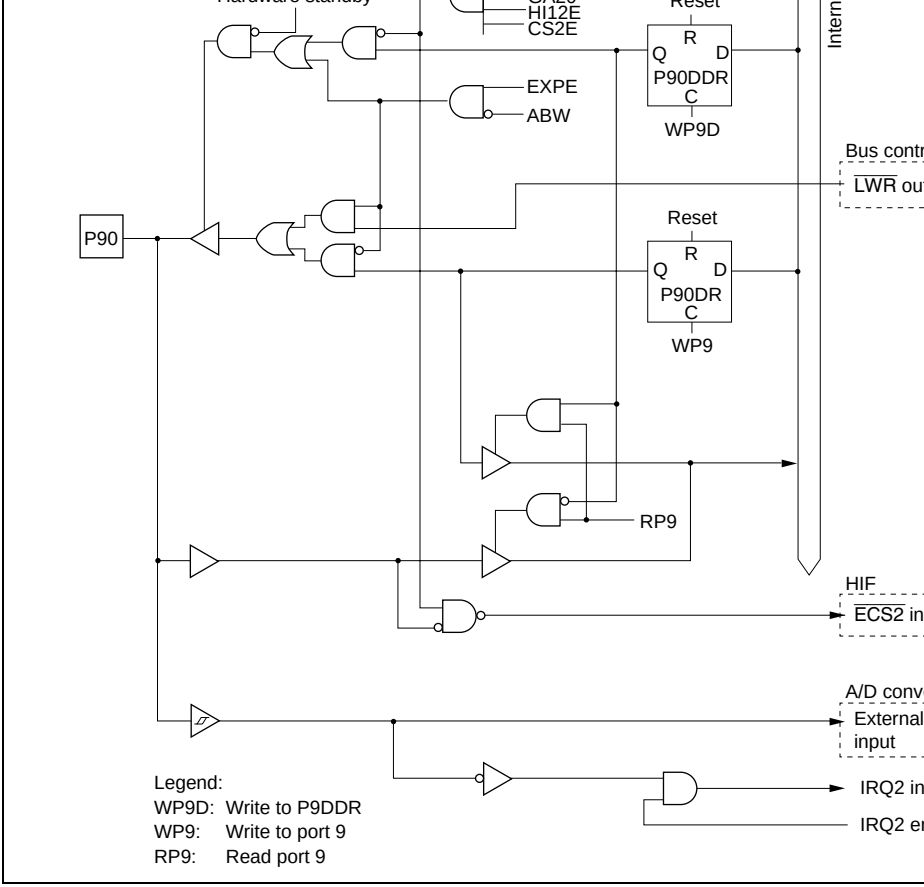


Figure C.29 Port 9 Block Diagram (Pin P90)

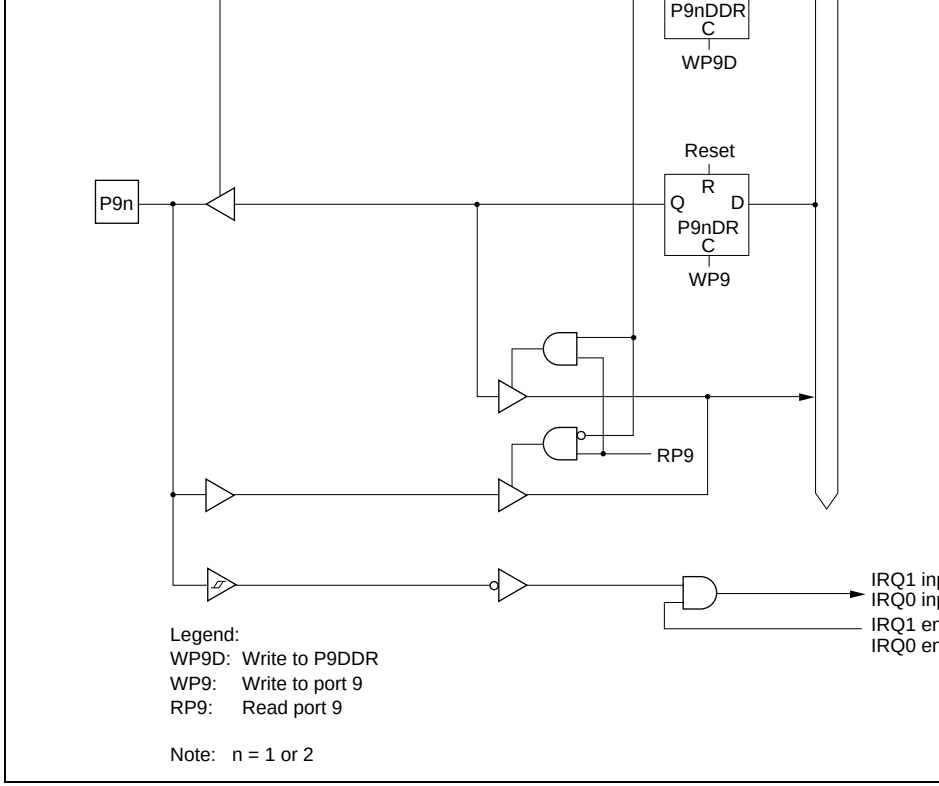


Figure C.30 Port 9 Block Diagram (Pins P91, P92)

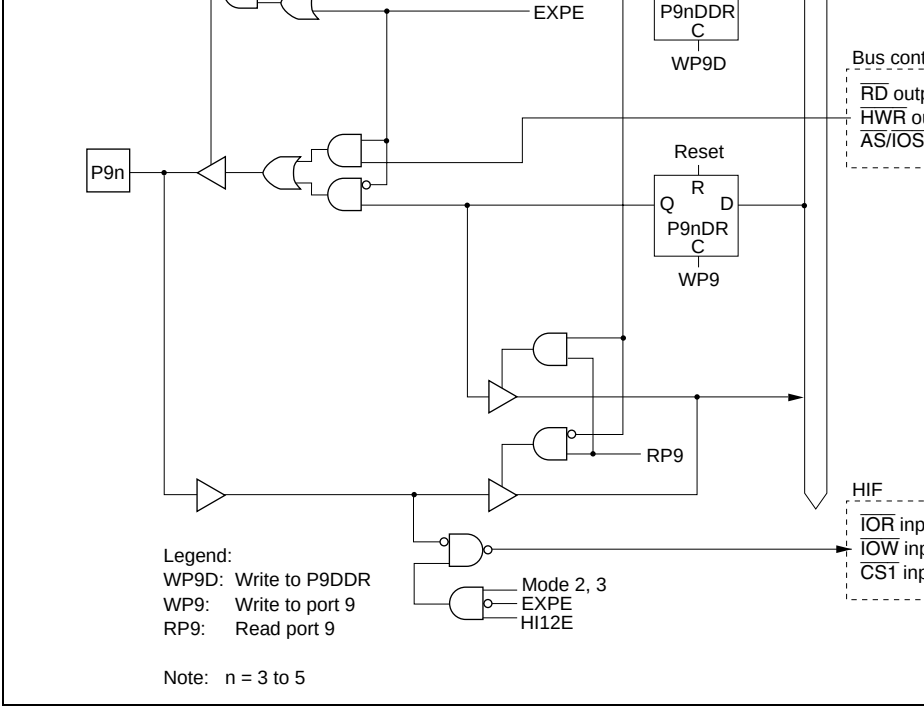


Figure C.31 Port 9 Block Diagram (Pins P93 to P95)

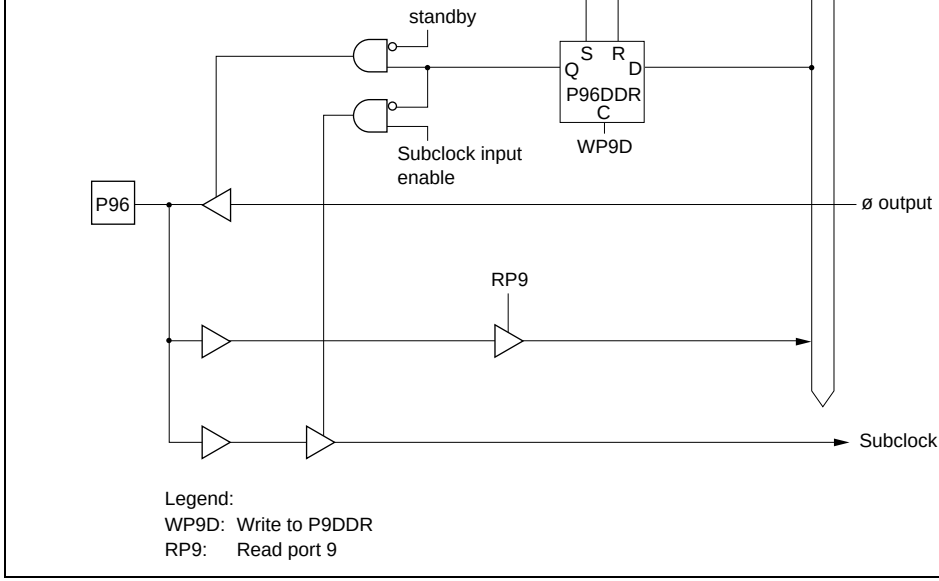
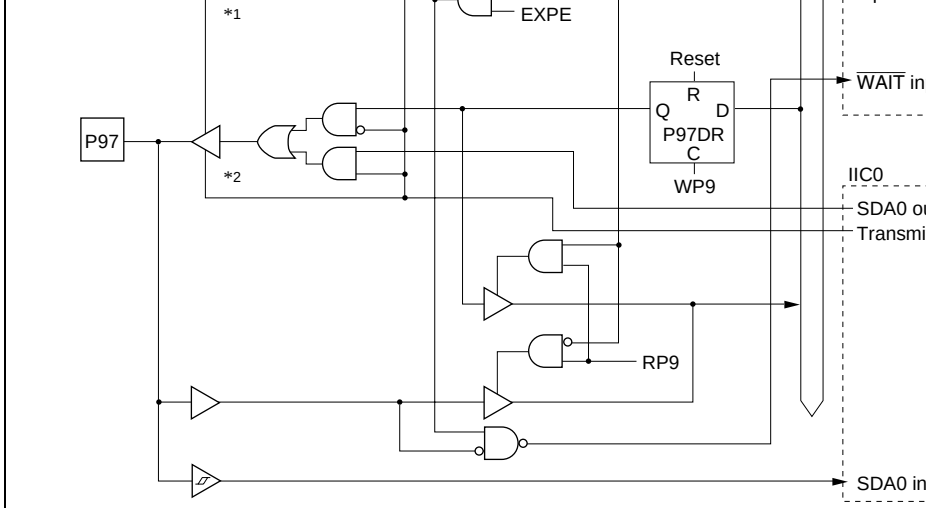


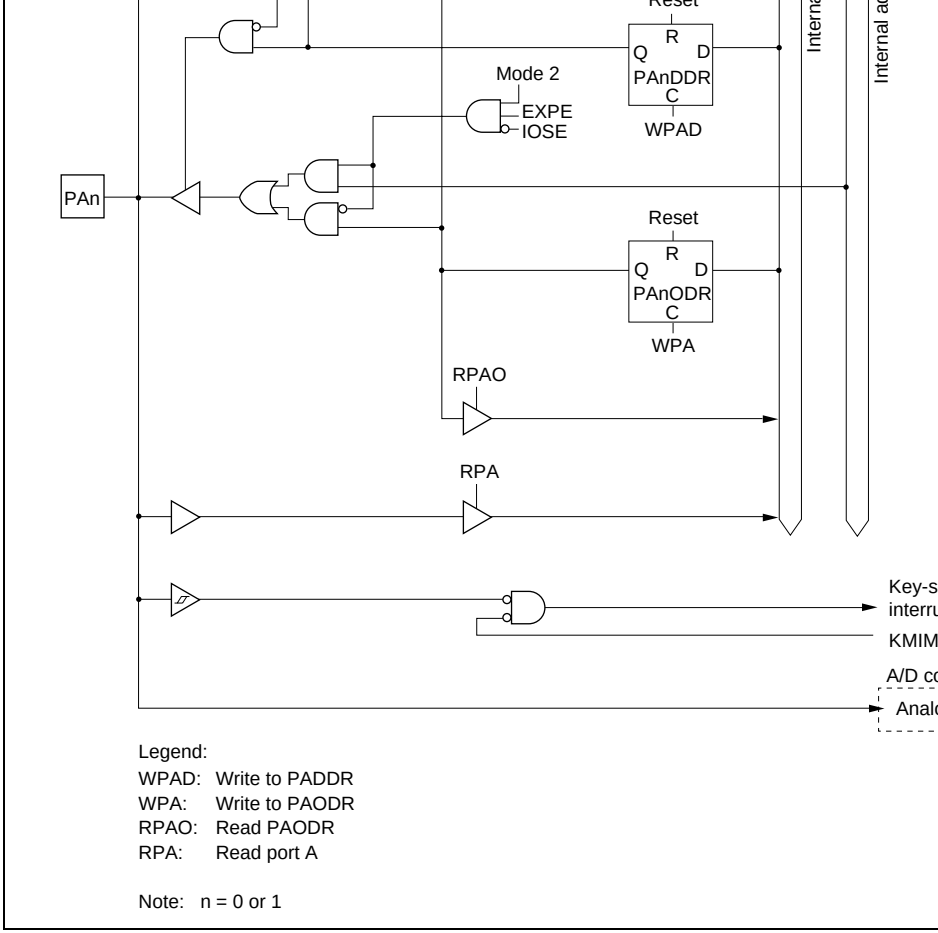
Figure C.32 Port 9 Block Diagram (Pin P96)



Legend:
WP9D: Write to P9DDR
WP9: Write to port 9
RP9: Read port 9

Notes: 1. Output enable signal
2. Open drain control signal

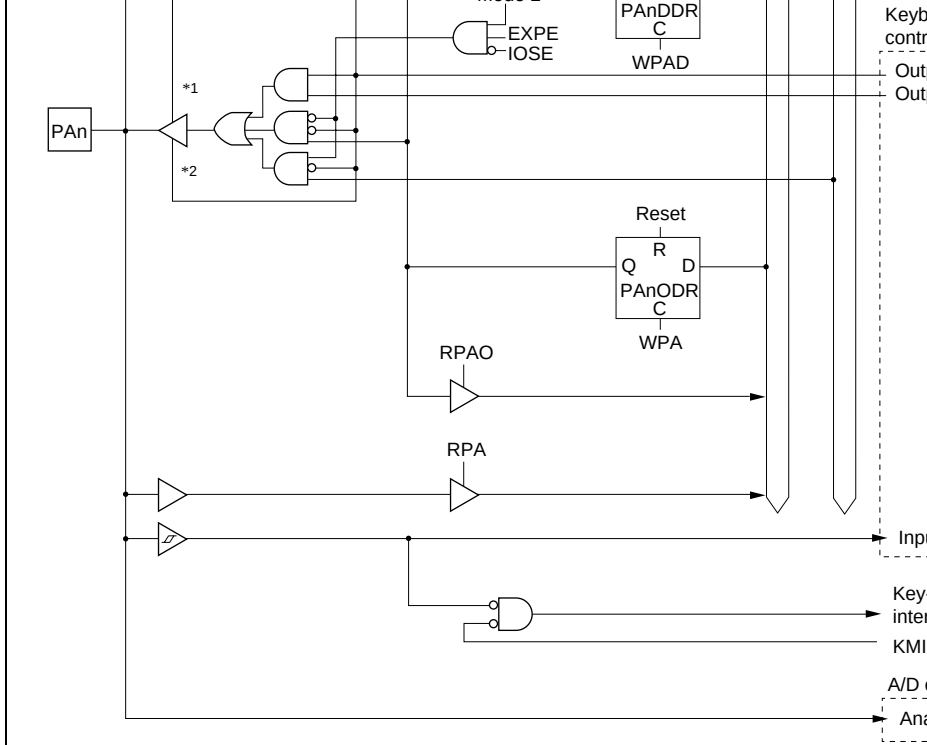
Figure C.33 Port 9 Block Diagram (Pin P97)



Legend:
WPAD: Write to PADDR
WPA: Write to PAODR
RPAO: Read PAODR
RPA: Read port A

Note: n = 0 or 1

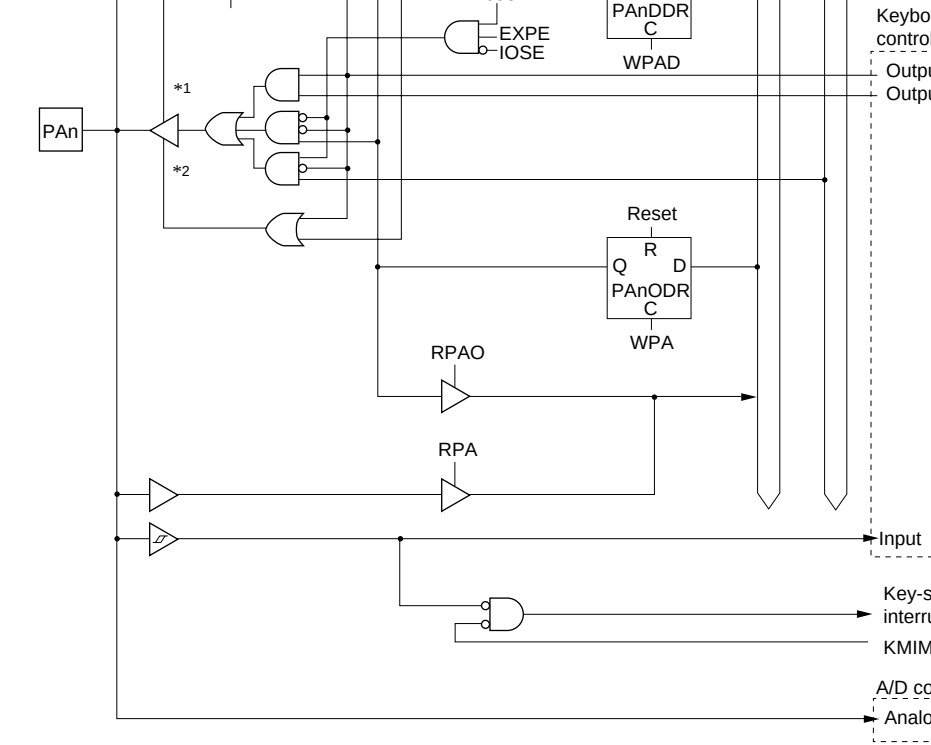
Figure C.34 Port A Block Diagram (Pins PA0, PA1)



Legend:
WPAD: Write to PADDR
WPA: Write to PAODR
RPAO: Read PAODR
RPA: Read port A

Notes: n = 2 or 3
1. Output enable signal
2. Open-drain control signal

Figure C.35 Port A Block Diagram (Pins PA2, PA3)



Legend:
WPAD: Write to PADDR
WPA: Write to PAODR
RPAO: Read PAODR
RPA: Read port A

Notes: n = 4 to 7
1. Output enable signal
2. Open-drain control signal

Figure C.36 Port A Block Diagram (Pins PA4 to PA7)

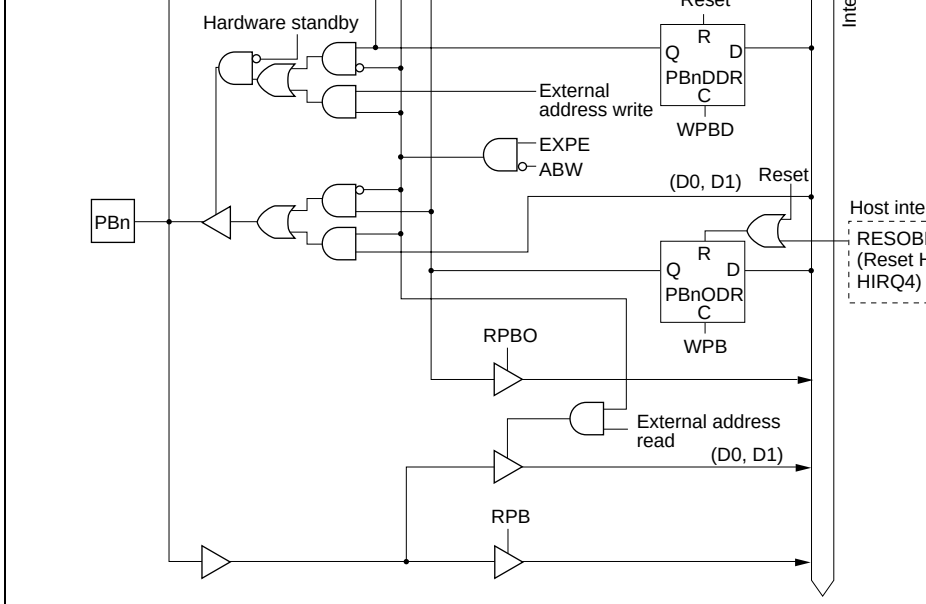


Figure C.37 Port B Block Diagram (Pins PB0 and PB1)

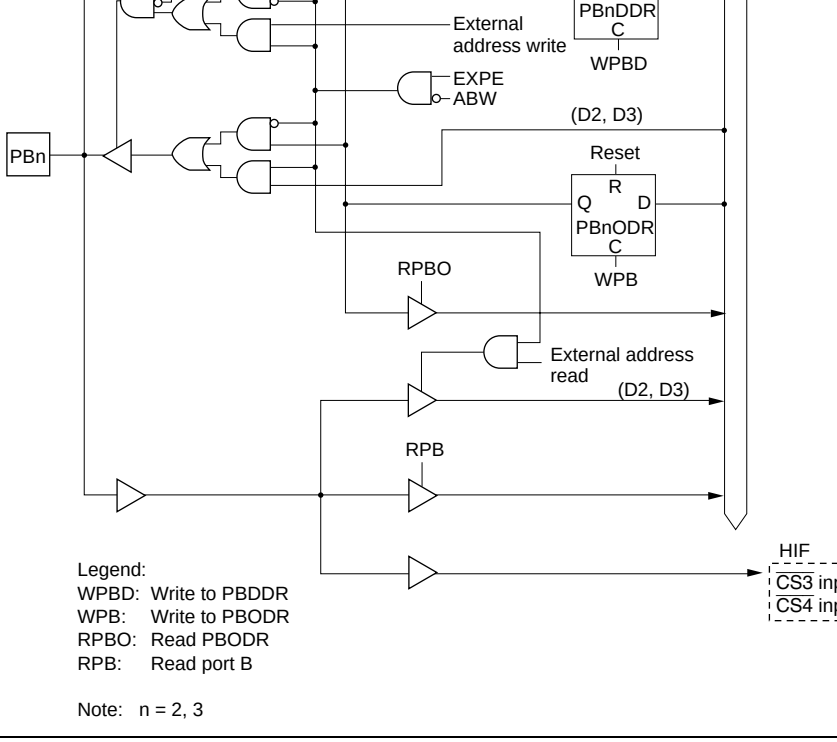
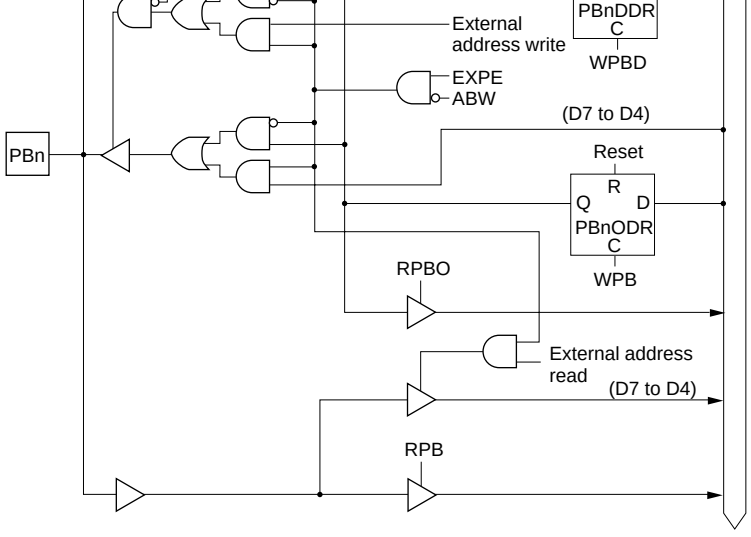


Figure C.38 Port B Block Diagram (Pins PB2 and PB3)



Legend:
WPBD: Write to PBDDR
WPB: Write to PBODR
RPBO: Read PBODR
RPB: Read port B

Note: n = 4 to 7

Figure C.39 Port B Block Diagram (Pins PB4 to PB7)

Pin Name	Mode	Reset	Mode	Mode	Mode	Mode	Mode	Mode
Port 1 A7 to A0	1	L	T	keep*	keep*	keep*	keep*	A7 to A0
	2, 3 (EXPE = 1)	T						Address output/ input port
	2, 3 (EXPE = 0)							I/O port
Port 2 A15 to A8	1	L	T	keep*	keep*	keep*	keep*	A15 to A8
	2, 3 (EXPE = 1)	T						Address output/ input port
	2, 3 (EXPE = 0)							I/O port
Port 3 D15 to D8	1	T	T	T	T	T	T	D15 to D8
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)			keep	keep	keep	keep	I/O port
Port 4	1	T	T	keep	keep	keep	keep	I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 5	1	T	T	keep	keep	keep	keep	I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 6	1	T	T	keep	keep	keep	keep	I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 7	1	T	T	T	T	T	T	Input port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 8	1	T	T	keep	keep	keep	keep	I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 97 $\overline{\text{WAIT}}$	1	T	T	T/keep	T/keep	T/keep	T/keep	$\overline{\text{WAIT}}$/ I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)			keep	keep	keep	keep	I/O port

	2, 3 (EXPE = 0)			keep	keep	keep	keep	I/O port
Port 92 to 91	1	T	T	keep	keep	keep	keep	I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)							
Port 90 $\overline{\text{LWR}}$	1	T	T	H/keep	H/keep	H/keep	H/keep	$\overline{\text{LWR}}$/ I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)			keep	keep	keep	keep	I/O port
Port A A23 to A16	1	T	T	keep*	keep*	keep*	keep*	I/O port
	2, 3 (EXPE = 1)							A23 to A16/ I/O port
	2, 3 (EXPE = 0)							I/O port
Port B D7 to D0	1	T	T	T/keep	T/keep	T/keep	T/keep	D7 to D0/ I/O port
	2, 3 (EXPE = 1)							
	2, 3 (EXPE = 0)			keep	keep	keep	keep	I/O port

Legend:

H: High

L: Low

T: High-impedance state

keep: Input ports are in the high-impedance state (when DDR = 0 and PCR = 1, MOS
ups remain on).

Output ports maintain their previous state.

Depending on the pins, the on-chip supporting modules may be initialized and t
function determined by DDR and DR used.

DDR: Data direction register

Note: * In the case of address output, the last address accessed is retained.

remain low until $\overline{STBY}$ signal goes low (minimum delay from $\overline{STBY}$ low to $\overline{RES}$ h

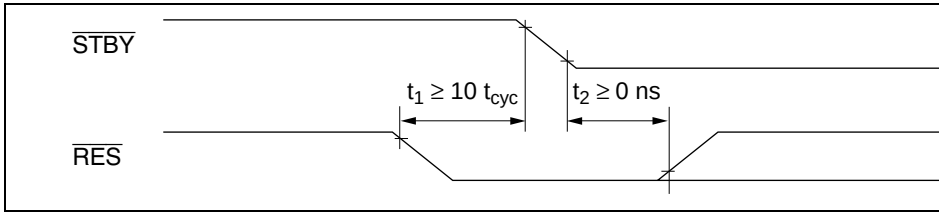


Figure E.1 Timing of Transition to Hardware Standby Mode

(2) To retain RAM contents with the RAME bit cleared to 0 in SYSCR, or when RAM not need to be retained, $\overline{RES}$ does not have to be driven low as in (1).

E.2 Timing of Recovery from Hardware Standby Mode

Drive the $\overline{RES}$ signal low at least 100 ns before $\overline{STBY}$ goes high to execute a reset.

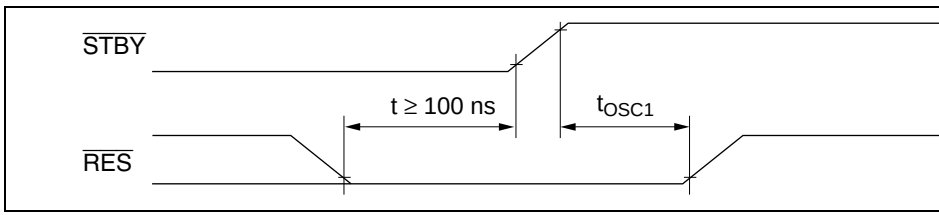


Figure E.2 Timing of Recovery from Hardware Standby Mode

						(TFP-1
			Version with on-chip I²C bus interface (5-V version, 4-V version, 3-V version)	HD6432148SW	HD6432148S(V)W(***)FA	100-pin (FP-10
					HD6432148S(V)W(***)TE	100-pin (TFP-1
		F-ZTAT version	Standard product (5-V version/4-V version)	HD64F2148	HD64F2148FA20	100-pin (FP-10
					HD64F2148TE20	100-pin (TFP-1
			Low-voltage version (3-V version)	HD64F2148V	HD64F2148VFA10	100-pin (FP-10
					HD64F2148VTE10	100-pin (TFP-1
	H8S/2147	Mask-ROM version	Standard product (5-V version, 4-V version, 3-V version)	HD6432147S	HD6432147S(V)(***)FA	100-pin (FP-10
					HD6432147S(V)(***)TE	100-pin (TFP-1
			Version with on-chip I²C bus interface (5-V version, 4-V version, 3-V version)	HD6432147SW	HD6432147S(V)W(***)FA	100-pin (FP-10
					HD6432147S(V)W(***)TE	100-pin (TFP-1
H8S/2148 Group A-mask version	H8S/2148A	F-ZTAT version A-mask version	Standard product (5-V version/4-V version)	HD64F2148A	HD64F2148AFA20	100-pin (FP-10
					HD64F2148ATE20	100-pin (TFP-1
			Low-voltage version (3-V version)	HD64F2148AV	HD64F2148AVFA10	100-pin (FP-10
					HD64F2148AVTE10	100-pin (TFP-1
	H8S/2147A	F-ZTAT version A-mask version	Standard product (5-V version/4-V version)	HD64F2147A	HD64F2147AFA20	100-pin (FP-10
					HD64F2147ATE20	100-pin (TFP-1
			Low-voltage version (3-V version)	HD64F2147AV	HD64F2147AVFA10	100-pin (FP-10
					HD64F2147AVTE10	100-pin (TFP-1

H8S/2144 Group	H8S/2144	Mask-ROM version	Standard product (5-V version, 4-V version, 3-V version)	HD6432144S	HD6432144S(V)(***)FA	100-pin Q (FP-100E
					HD6432144S(V)(***)TE	100-pin T (TFP-100
		F-ZTAT version	Standard product (5-V version/4-V version)	HD64F2144	HD64F2144FA20	100-pin Q (FP-100E
					HD64F2144TE20	100-pin T (TFP-100
			Low-voltage version (3-V version)	HD64F2144V	HD64F2144VFA10	100-pin Q (FP-100E
					HD64F2144VTE10	100-pin T (TFP-100
	H8S/2143	Mask-ROM version	Standard product (5-V version, 4-V version, 3-V version)	HD6432143S	HD6432143S(V)(***)FA	100-pin Q (FP-100E
					HD6432143S(V)(***)TE	100-pin T (TFP-100
	H8S/2142	Mask-ROM version	Standard product (5-V version, 4-V version, 3-V version)	HD6432142	HD6432142(***)FA	100-pin Q (FP-100E
					HD6432142(***)TE	100-pin T (TFP-100
		F-ZTAT version	Standard product (5-V version/4-V version)	HD64F2142R	HD64F2142RFA20	100-pin Q (FP-100E
					HD64F2142RTE20	100-pin T (TFP-100
			Low-voltage version (3-V version)	HD64F2142RV	HD64F2142RVFA10	100-pin Q (FP-100E
					HD64F2142RVTE10	100-pin T (TFP-100
H8S/2144 Group A-mask version	H8S/2144A	F-ZTAT version A-mask version	Standard product (5-V version/4-V version)	HD64F2144A	HD64F2144AFA20	100-pin Q (FP-100E
					HD64F2144ATE20	100-pin T (TFP-100
			Low-voltage version (3-V version)	HD64F2144AV	HD64F2144AVFA10	100-pin Q (FP-100E
					HD64F2144AVTE10	100-pin T (TFP-100

Note: (***) is the ROM code.
The F-ZTAT version of the H8S/2148 has an on-chip I^2C bus interface as standard.
The F-ZTAT 5 V/4 V version supports the operating ranges of the 5 V version and the 4 V
The operating range of the F-ZTAT low-voltage version will be decided later.
The above table includes products under development. Information on the status of individ
can be obtained from Renesas sales offices.

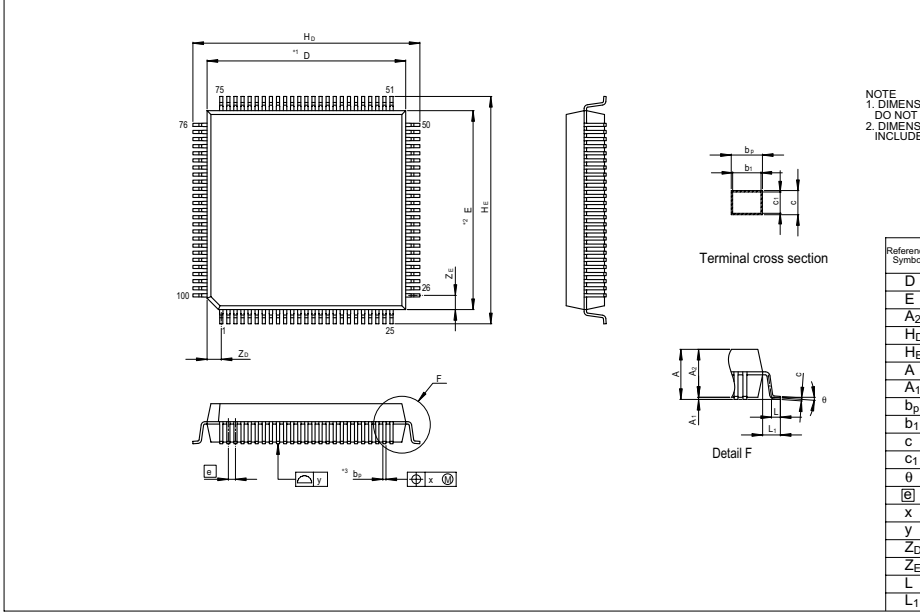
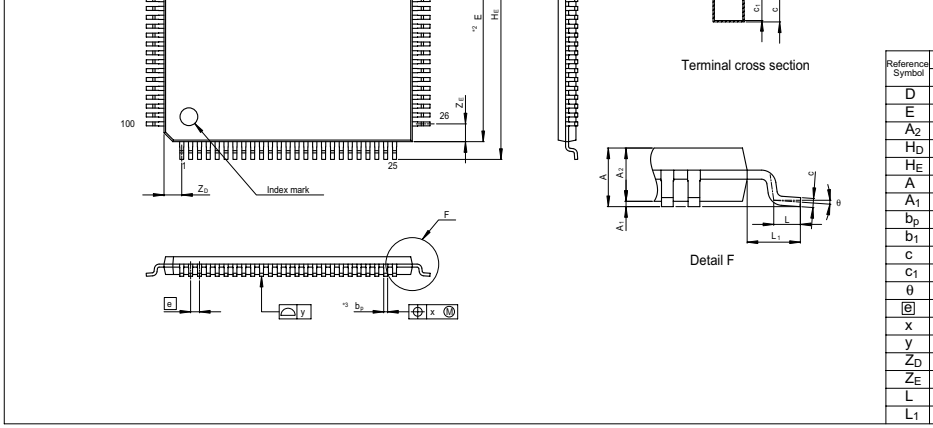


Figure G.1 Package Dimensions (FP-100B)



Reference Symbol
D
E
A_2
H_D
H_E
A
A_1
b_p
b_1
c
c_1
θ
e
x
y
Z_D
Z_E
L
L_1

Figure G.2 Package Dimensions (TFP-100B)

Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8S/2148 Group, H8S/2144 Group,
H8S/2148FZTAT™, H8S/2147N F-ZTAT™,
H8S/2144F-ZTAT™, H8S/2142F-ZTAT™

Publication Date:	1st Edition, November 1999
	Rev.4.00, September 27, 2006
Published by:	Sales Strategic Planning Div.
	Renesas Technology Corp.
Edited by:	Customer Support Department
	Global Strategic Communication Div.
	Renesas Solutions Corp.

RENESAS SALES OFFICES

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 Ehs
Tel: <603> 7955-9390, Fax: <603> 7955-9510

H8S/2148 Group, H8S/2144 Group, H8S/2148F-ZTAT™, H8S/2147N F-ZTAT™, H8S/2144F-ZTAT™, H8S/2142F-ZTAT™

Hardware Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0327-0400

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 :

MB90F036APMC-GSE1 MB90F342CASPMC-GSE1 MB90F345CESPMC-GE1 MB90F349CAPFR-GSE1 MB90F428GCPFR-GSE1 MB90F462APFM-GE1 MB90F462APMC-G-SNE1 MB90F497GPF-GE1 MB90F546GSPFR-GE1 MB90F947APFR-GS-SPE1 MB96F346RSBPMC-GS-N2E2 MB96F683RBPMC-GSAE1 R5F11BGEAFB#30 DF3026XBL25V S912ZVFP64F1VLL R4F24268NVRFQV R5F107DEGSP#X0 R5F11B7EANA#U0 R5F21172DSP#U0 M30622F8PGP#U3C MB90092PF-G-BNDE1 MB90F335APMC1-G-SPE1 MB90F342CASPFR-GS-N2E1 MB90F345CAPFR-GSE1 MB90F543GPF-GE1 MB90F546GSPF-GE1 MB90F568PMCR-GE1 MB90F594APFR-GE1 MB90F882ASPMC-GE1 MB96F346RSAPQCR-GS-N2E2 MB96F387RSBPMC-GSE2 MB96F387RSBPMC-GS-N2E2 MB96F395RSAPMC-GSE2 MB96F623RBPMC1-GSE1 MB96F646RBPMC-GSE1 XE167F96F66LACFXUMA1 MB96F696RBPMC-GSAE1 MB96F018RBPMC-GSE1 MB90F962SPMCR-GE1 MB90F867ASPFR-GE1 MB90F543GPF-G-FLE1 MB90F345CESPF-GE1 M30290FCHP#U3A DF2239FA20IV HD64F3672FPV R5F104AEASP#V0 R5F100BCANA#U0 R5F100BFANA#U0 S9S12H256J2VFVER R5F100ACASP#V0